ISSN 0769-3836

N° 10 JUIN-JUILLET 86 PRIX 20 F

exelement vôtre



- informatique et musique
- EXL 100 reportage dans une école
- banc d'essai: ATI 42
- réalisez des cartes d'extension pour votre EXL 100

EDITO
Musique et informatique

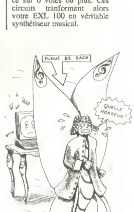
A l'heure actuelle, les ordinateurs dits musicaux occupent une place de choix dans l'univers du musicien professionnel, ou amateur. Et oui, les octets remplacent peu à peu les guitares, pianos et autres instruments. Comment cela est-il possible ? Simplement par la numérisation ou la digitalisation des sons. Si ces termes vous semblent barbares, les techniques employées pour arriver au résultat désiré, à savoir restituer le timbre d'un instrument, ne le sont pas moins. Et, pour cette raison, nous ne les aborderons pas.

L'informatique au service du musicien amateur.

Si vous n'êtes pas un amateur éclairé de la musique, les ordinateurs musicaux (ordinateur + logiciel spécifique) n'auront aucun intérêt pour vous, sauf si vous désirez jouer de la musique sans investir du temps à l'apprentissage du solfège, et d'une technique instrumentale.En effet, il est possible avec un ordinateur de jouer une fugue de Bach, sans jamais avoir appris une seule note. Votre travail consistera à copier note à note une partition en utilisant un logiciel spécifique. Mais, ces logiciels coûtent chers, surtout si vous désirez avoir une qualité d'exécution proche de la réalité, et de plus, ils ne sont pas disponibles, loin s'en faut, sur toutes les machines. Alors que reste-t-il au pauvre malheureux acheteur qui croyait posséder en la personne de son EXL 100 un exécutant de la qualité d'un Weissemberg ou autre ? Pas mal de choses si l'on considère le besoin moyen des utilisateurs de micro-ordinateurs domestiques pour ce qui concerne la musique.

Il se peut que vous soyez un novice en musique. De ce fait, vous pouvez utiliser votre EXL 100 comme un répétiteur de musique. Les logiciels MUSIQUE1 et MUSIQUE2 ont comme fonction principale la dictée musicale, et ils vous apprendront les différentes notes et leurs emplacements sur la portée en fonction de la clé utilisée (clé de sol ou clé de fa). Ces logiciels constituent un bon investissement de base pour la personne qui désire s'initier au solfège. Et puis, avouons-le, il est plus amusant d'apprendre le solfège sur un ordinateur, plutôt que de passer par le circuit traditionnel. (professeur de musique, conservatoire et la méthode rose en cadeau).

Une fois que vous aurez fait vos premiers pas dans le monde de la musique, il se peut que vous ayez quelques velléités de création musicale. Deux solutions s'offrent à vous. Apprendre une technique instrumentale ou utiliser l'ordinateur. La première solution est de loin celle qui vous coûtera le plus de temps. Apprendre un instrument n'est pas chose facile, et de plus, l'apprentissage sur un instrument quelconque est un travail ingrat pour l'exécutant, et un véritable supplice pour les voisins. Le logiciel REMI sur EXL 100 permet la création de lignes mélodiques. Ces lignes mélodiques peuvent être ensuite intégrées dans un programme BASIC. Ce générateur de mélodies vous permet de jouer sur 3 octaves et une voie. L'EXL 100 ne permet la génération de notes que sur une seule voie. Mais, il est possible en utilisant le bus domestique de coupler un circuit imprimé spécialisé dans le traitement des informations musicales, permettant ainsi le génération de notes sur 5 ou 6 octaves et ce sur 6 voies ou plus. Ces circuits tranforment alors votre EXL 100 en véritable synthétiseur musical.





Si vous êtes un pro ou un semi-pro de la musique, votre EXL 100 ne pourra pas vous être utile à grand chose, sauf si vous utilisez Exeldrums. Un petit mot sur Exeldrums. Cette boîte à rythmes est actuellement en promotion à 690, 00 Frs chez vos revendeurs. Pour ce prix là, vous avez une boîte à rythmes de qualité professionnelle. La revue SONO spécialisée dans les techniques sonores a publié au mois de septembre 1985, un banc d'essai élogieux sur Drums. Etant musicien à mes moments perdus, j'utilise Drums dans toutes les parties percussions en studio d'enregistrement pour la réalisation d'indicatifs ou de chansons. Sa facilité d'emploi permet une utilisation très souple. A part Exeldrums, les logiciels professionnels de musique n'existent pas sur EXL 100.

MIDI ou la standardisation réussie

Il reste un aspect des choses à analyser. Un ordinateur peut être utilisé pour contrôler des synthétiseurs, des boîtes à rythmes, ou des séquenceurs. Cette particularité des ordinateurs n'est apparue que ces deux dernières années. En effet, les constructeurs mondiaux de matériel de musique ont décidé, il y a un peu plus de deux ans, d'équiper tous les instruments de musique électronique d'une interface. Cette interface MIDI (Musical Interface for Digital Instruments) permet de faire communiquer tous les intruments qui en sont pourvus. De cette façon, un synthétiseur de marque YAMAHA peut émuler un clavier ROLAND, et vice versa. Cette interface permet d'adapter des périphériques aux instruments que vous possédez et donc d'étendre et ce, à moindre frais, votre synthétiseur. Il est regrettable que les constructeurs d'ordinateurs, n'imitent pas leurs collègues en adoptant un standard unique, pour le plus grand profit des utilisateurs. Mais bien souvent, le profit des utilisateurs ne rime pas avec le profit des constructeurs.

De ce fait, la musique est devenue pratiquement une affaire d'ordinateurs et de gros sous. La qualité de l'instrumentiste importe peu, les ordinateurs suppléent à la médiocrité. Certes, les artistes de variétés célèbres utilisent des ordinateurs sur scène pour recréer le son que l'on ne pouvait jadis qu'obtenir en studio d'enregistrement, et à grands frais. Mais ne rêvez pas, pour avoir l'illusion de ressemblez à Jean-Jacques Goldman ou Michel Berger, il vous faudra dégager un budget d'au moins cent mille francs, ce qui n'est pas à la portée de Monsieur tout le monde.

Si vous posez la question piège: Pourquoi l'EXL 100 ne possède pas d'interface MIDI ?,Nous pouvons répondre du tac au tac: Saviez-vous qu'une telle interface existait et connaissiez-vous son rôle ? En fait, l'interface MIDI est destinée aux professionnels de la musique (Certes des amateurs l'utilisent, mais son utilité réelle se situent au niveau de la scène et du studio d'enregistrement). Néanmoins il est possible de construire une interface MIDI et de l'adapter sur un EXL 100. Jean Claude Repetto nous proposera dans un prochain numéro le schéma de montage de cette interface. Le coût de cette dernière est de l'ordre de 100 francs environ. Nous prévoyons même la possibilité d'interfacer un circuit spécialisé pour la création musicale si la demande des utilisateurs d'EXL 100 est importante.





L'avenir des ordinateurs musicaux

La dernière heure des instruments conventionnels a-t-elle sonné ? A l'heure actuelle, NON. Loin s'en faut. Si l'ordinateur est accepté par de plus en plus de personnes, il semble illusoire d'imaginer que l'on jouera à Playel le concerto numéro 21 de W. A. Mozart pour ordinateur et séquenceur. (Les vieilles barbes attraperaient un infarctus) . Toutefois, il n'est pas impossible de penser que l'apprentissage de la musique soit réalisé dans les classes primaires avec des ordinateurs. Pour ce qui concerne l'aspect visuel d'une représentation musicale, voir une personne tapoter du bout des doigts sur un clavier d'ordinateur n'est guère spectaculaire , et donc voué à l'échec. Seuls les studios d'enregistrement sont en mesure de travailler avec des ordinateurs pour fabriquer en secret le "son" du dernier album de "DIRE STRAITS", ou de PHIL COLLINS. Les créateurs de sons trouvent dans l'ordinateur l'instrument idéal pour forger de nouvelles sonorités. La grande révolution de l'informatique musicale c'est le sampler, ou l'échantilloneur, qui permet d'analyser, et donc de reproduire une quelconque sonorité. Du bruit d'une motocyclette, aux ondes délicates d'une flûte de Pan, l'échantilloneur ouvre la porte aux pionners des nouvelles sonorités.

Nouvelles brèves

Dernier numéro d'Exelement vôtre: La direction d'Exelvision a décidé que le dernier numéro d'Exelement vôtre sera le numéro 15. Pour cette raison, le bulletin habituel d'abonnement est supprimé au profit d'un bulletin de souscription qui permettra aux lecteurs ne possédant pas les premiers numéros d'EV de se les procurer. Pourquoi cette décision a-t-elle été prise?
Ce n'est un secret pour personne que nous nous orientons vers un service télématique. En effet, nous serons en mesure dès le mois d'octobre 1986 de proposer à nos lecteurs et aux acheteurs de matériel Exelvision, une revue télématique. En conséquence de quoi, il semble absurde de continuer la revue sur support papier. Cette dernière continuera jusqu'au mois de Mai 1987 pour assurer la transition. De plus, dans une certaine mesure, la revue Exelement vôtre sera un corollaire du service télématique Surtout, ne croyez pas que cette revue soit arrêtée par manque d'abonnés, non, nous enregistrons plus de quatre mille abonnés. Mais, il est de plus en plus difficile pour une seule personne de pouvoir concevoir, écrire, mettre en page la revue. Soucieux de la qualité à apporter à nos produits, nous préférons passer la main à d'autres confrères qui se chargeront de distribuer et écrire une nouvelle revue. Cette dernière ne portera plus le nom d'Exelement Vôtre, et sera distribuée dans les kiosques.



Exelclub. L'Exelclub change de locaux et offre maintenant un service de formation. Les nouveaux locaux sont situés 40 Rue Davy, 75017 Paris. Tel 16 / 1 42298001. Les possesseurs de matériel Exelvision pourront se perfectionner, ou apprendre à manipuler le matériel. Ah, les heures de fonctionnement du club sont de 10h à 20h, tous les jours.

Exeltexte sur disquettes

Les possesseurs de l'unité de disquettes EXL 135 ont du rencontrer quelques difficultés pour sauvegarder leurs textes créés à partir d'Exeltexte. En effet, la version de base d'Exeltexte n'est pas prévue pour fonctionner avec l'unité de disquettes. Toutefois, une version actualisée sur disquettes est envoyée contre la cartouche ROM Si vous

désirez obtenir la version disquette du logiciel Exeltexte, renvoyez votre cartouche ROM à Exelvision, Place Joseph Bermond, immeuble Ophira, 06560.

Schéma de l'interface magnétophone

Dans le dernier numéro d'Exelement Vôtre, nous avons publié le schéma de l'interface magnétophone dans une échelle plus petite que 1. Nous vous demandons de nous excuser. Vous trouverez dans ce numéro le schéma à l'échelle exacte. Les deux schémas recto et verso de l'interface sont publiés sur une page entière. En effet, il est possible grâce à une bombe de produit spécial de rendre le papier non imprimé transparent. Ceci permet de réaliser directement la plaque de montage.

Note aux écoles

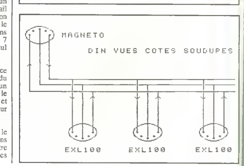
Existe-t-il une solution moins coûteuse que le Nanoréseau LEANORD (64 000,00 Frs) pour équiper six postes de travail dans une classe ?
Oui!!!
Il est possible d'associer un EXL 100 (+ MODEM+ EXELMEMOIRE) connecté au réseau télétel, à un ensemble de postes de travail (EXL 100 en version standard). En outre, le système que nous décrivons ci-après permet de charger 7 EXL 100 à partir d'un seul magnétophone.

Dans un premier temps ce dispositif permet à partir du poste maître de télécharger un logiciel disponible sur le centre serveur Exelvision, et de le stocker sur Exelmémoire.

Dans un second temps, le logiciel stocké dans l'Exelmémoire peut être distribué via les interfaces cassettes des différents EXL 100.

Plus précisément, les différentes manipulations sont les suivantes:
-Téléchargement logiciel
-Mise en place de la cartouche BASIC sur le poste maître
-Chargement dans la mémoire de ce dernier du logiciel précédement téléchargé
-Mise en attente des postes esclaves en vue du chargement de ceux-ci:
-A partir d'un seul clavier on frappe l'ordre LOAD"1" [ENVOI] suivi de l'appui sur [ESC].
-Sur le poste maître frappez l'ordre SAVE "1.PROG" [ENVOI] puis [ESC]

A partir de cet instant le programme transite du poste maître vers les postes esclaves. Bien évidemment le poste maître peut servir de poste de travail, ce qui optimise l'utilisation des EXL 100 mis en jeu. Les mauvaises langues diront que le processus prend autant de temps que le chargement d'une cassette, l'avantage est que l'investissement en matériel de gestion des postes esclaves et de connectique est très faible voir nul. (Signalons pour mémoire que le coût des câbles pour l'installation d'un nanoréseau est de l'ordre de 16 000 Francs) Voici le schéma de raccordement des EXL 100 esclaves à l'EXL 100 maître:



De la même manière, et en vue d'économiser temps et cassettes,il est possible de charger plusieurs EXL 100 à partir d'un seul magnétophone (bien réglé). Il devient donc inutile de charger les postes l'un après l'autre, ou de dupliquer les cassettes originales. Le schéma du câble qui relie le magnétophone aux différents EXL 100 devient · (Voir page précédente)

Ce dernier dispositif a été testé positivement avec sept EXL 100 à partir d'une cassette commercialisée, et donc dupliquée de façon industrielle

En conclusion, cette ébauche de réseau rendra bien des services aux instituteurs, et ce, à un prix défiant toute concurrence: Possibilité d'envoyer depuis le poste maître un logiciel à un, ou plusieurs postes esclaves, et inversement, transférer un programme d'un poste esclave en direction du poste maître.



L 'école de Jules Ferry et l'ordinateur font-ils bon ménage ?

Nous sommes partis ce matin pour rencontrer un directeur d'école qui a reçu de l'Education Nationale la dotation Exelvision (1 EXL 100, 1 imprimante, 1 modem, et des brouilles). Le choix de ce directeur d'école n'est pas du au hasard. Non, pas du tout, Nicolas, ingénieur technico-commercial, un beau matin m'a déclaré: "Je connais un instit qui utilise 5 EXL 100 dans sa classe "

Bigre, fantastique 5 EXL 100 pour une classe, c'est du jamais vu. Je tenais vraiment à me rendre compte de l'exploit, et rendez-vous pris, nous voilà partis pour l'école.

L'école de Castagniers les Moulins située à la périphérie de Nice est une école dans le plus pur style laïque. Bâtisse solide, n'ayant rien à voir avec les préfabriqués modernes. Le directeur et instituteur de l'école, Monsieur Caréna nous accueille dans son fief. Méridional, cela s'entend à l'accent qui chante, il nous invite à le suivre dans la classe des grands. C'est une petite école, d'environ 70 élèves. Monsieur Caréna dirige une classe triple les CE2, CM1, et CM2. Nous entrons dans la classe, et là, bien rangés sur des plans de travail construits artisanalement trônent 5 EXL 100.

Image folle en fait que celle de Jules Ferry cohabitant avec celle de Bob Noyce (inventeur du microprocesseur). D'un côté l'aspect vieillot, presque suranné de cette classe: l'immense tableau noir, les cartes de géographie colorées, les chefs-d'oeuvre des enfants qui ornent les mûrs laiteux, et d'un autre côté, une rangée de micro-ordinateurs. En fait ça n'a pas beaucoup changé depuis que j'ai quitté la communale, à part que les



micros, l'on n'en avait jamais entendu parlé, et que les seuls ordinateurs que l'on connaissait, c'étaient ceux qui illustraient les Strange ou autres Cpt América. Tempus fugit. Non pas que je me sente âgé comme Mathusalem, mais le monde a bien changé depuis 68 (date de mon CM2, que les lecteurs se rassurent.).

Nous entrons dans le vif du sujet. Comment se fait-il qu'une si petite école soit en possession de 5 EXL 100? Monsieur Caréna nous explique qu'il a profité d'une solde exceptionnelle sur le matériel et sur les moniteurs. La mairie a bien entendu financé l'achat. Mais la grande astuce de Monsieur Caréna c'est de se servir des 5 EXL 100 comme d'un réseau. En effet, un EXL 100 sert de serveur. Quatre EXL 100 sont esclaves. Le processus fonctionne très bien, et de cette manière, il peut faire travailler 5 élèves à chaque cession.

Ce matin même nous essayons de charger 6 EXL 100 avec un EXL 100. Réussite totale. 7 EXL 100 peuvent être mis en réseau. Comment réagissent les élèves à l'enseignement assisté par ordinateur ? Ils adorent ce système, mais en fait, l'ordinateur ne peut



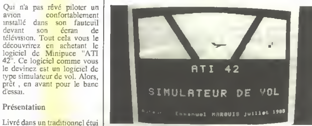
servir que comme support aux révisions, ou au contrôle des connaissances, l'enseignement étant réalisé de façon traditionnelle.



Dans cette classe l'EXL 100 est utilisé uniquement comme support pour les logiciels pédagogiques. Monsieur Caréna et son adjoint Monsieur Michelis n'enseignent pas la programmation à leurs élèves. Pourquoi? Ils n'ont pas été formés. Monsieur Caréna a bénéficié d'une semaine de formation en tout en pour tout sur TI 99 4A (offert par le conseil régional de la côte d'azur), quant à Monsieur

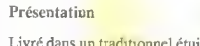
Michelis, il n'a bénéficié d'aucune formation. Cet état de fait ne doit pas être unique dans l'hexagone, et on est en droit de se poser une légitime question. Est-il raisonnable de dépenser des millions de francs dans des machines sans apporter la formation aux cadres qui doivent utiliser ces machine ? Le fait de ne pas maîtriser parfaitement les langages de programmation n'empêchent pas nos deux instituteurs de déployer un dynamisme des plus vivifiant pour ce qui concerne les nouvelles méthodes d'enseignement. Ils croyent fermement à l'EAO. L'EAO c'est une autre façon d'aborder les différents sujets de l'enseignement. Avant de se quitter Monsieur Caréna et Monsieur Michelis nous confiaient qu'ils envisageaient de passer à 10 le nombre d'EXL 100 de leur mini réseau. Gageons que cette bonne volonté de faire progresser l'EAO fasse des émules. N'oublions pas que derrière l'EAO se cache un pari fantastique: Former nos enfants aux tecniques de l'an 2000. N'oublions pas que les enjeux économiques de demain seront gagnés par nos enfants, alors donnons leurs tous les atouts pour réussir.

SIMULATEUR DE VOL ATI 42



Qui n'a pas rêvé piloter un avion confortablement installé dans son fauteuil devant son écran de télévision. Tout cela vous le découvrirez en achetant le logiciel de Minipuce "ATI 42". Ce logiciel comme vous le devinez est un logiciel de type simulateur de vol. Alors, prêt , en avant pour le banc d'essai.

Présentation

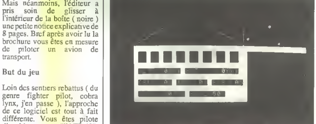


Livré dans un traditionnel étui du format vidéocassette, le logiciel ATI 42 n'est pas prodigue en documentation. Mais néanmoins, l'éditeur a pris soin de glisser à l'intérieur de la boîte (noire) une petite notice explicative de 8 pages. Bref après avoir lu la brochure vous êtes en mesure de piloter un avion de transport.

But du jeu

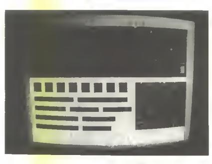
Loin des sentiers rebattus (du genre fighter pilot, cobra lynx, j'en passe), l'approche de ce logiciel est tout à fait différente. Vous êtes pilote d'un bi-moteur, "l'ATI 42", et votre mission consiste à transporter des passagers sur le territoire français.

Le constructeur nous livre les caractéristiques de l'ATI 42. Pour les novices du simulateur de vol, sachez



justement qu'un simulateur de vol doit respecter les caractéristiques techniques de l'appareil. Cette fois-ci vous n'êtes pas assis dans un mirage, mais dans un honnête avion de transport.

L'ATI 42 est un avion de transport bi-moteur propulsé par des moteurs de 1800 chevaux, et pouvant emporter 50 passagers. Sa vitesse de décolage est de 120 noeuds, sa vitesse d'atterrissage est de



150 noeuds, la vitesse ascensionnelle d'atterrissage est de -2,00 ms-1.

Si vous vous sentez pousser des ailes c'est bon signe.

Le poste de pilotage

Le poste de pilotage est composé de cinq parties distinctes.

-La vision du pilote
-Le radar de navigation
-La direction et la force du vent
-Tableau de bord
-Plan de vol

La vision du pilote:

Elle est réduite, mais suggère néanmoins certains éléments, notamment la piste d'atterrissage. Cette partie montre aussi la ligne d'horizon (ou l'horizon artificiel).

Le radar de navigation:

Cette fonction est très utile, et permet de suivre la route de l'appareil, et indique les différents terrains d'atterrissage. De plus lorsque l'avion entre dans la zone d'un aéroport, le premier plan est remplacé par un système d'atterrissage automatique (radar d'approche), permettant au pilote amateur de centrer son appareil.

La direction et la force du vent:

Deux voyants pour indiquer la force et le sens du vent.

Le tableau de bord

Le tableau de bord renseigne le pilote sur tous les éléments essentiels. Une première rangée de voyants indique l'état de l'appareil.

-Train d'atterrissage
-Volets
-Becs
-Aérofreins
-Freins
-Systèmed'approche automatique
-Chronomètre

Tout ce qui se trouve dessous les sept voyants renseignent le pilote, sur:

La vitessse de l'avion
Le nombre de tours du moteur
La vitesse ascensionnelle
Le cap
L'inclinaison
Le roulis

Tout ce qui est nécessaire pour piloter un appareil est indiqué.

Premier contact

Autant vous dire que des simulateurs j'en ai essayé un max. Je me suis dis encore une m... à tester comme d'habitude. J'attendais donc la fin du chargement l'esprit en éveil à la moindre faute, et emprunt de la joyeuse humeur du lundi matin (tell me why I don't like mondays). Et alors là, miracle, ce simulateur m'a plutôt détendu. En effet, il est agréable de s'évader un tant soit peu, par la magie d'un écran, d'un bureau trop étroit , même si l'on habite la côte d'azur, même si l'on travaille à Sophia Antipolis, et même si le soleil est au rendez-vous. C'est sûr, ça ne supporte pas

la comparaison avec un Flight simulator II d'IBM, mais c'est très honnête pour le rapport qualité prix. Les commandes sont un peu lentes, mais le but du jeu est d'apprendre à piloter un appareil, et non à courser le Soviet, ou le Yankee dans les cieux azurés, et radioactifs de notre Europe occidentale.

Côté décolage, je me débrouille pas mal, et donc j'ai réussi à enlever l'appareil de la piste. Ah ça c'est toujours un pied que de voir la piste séloigner et l'horizon s'approcher. J'ai choisi de rallier Paris Roissy à Paris Orly 42 Km, pas de vent, la ballade en somme. Ouais, il faut le dire vite Si vous êtes capable de vous orienter sur un carte, vous n'aurez aucun problème. Personnellement j'aurai des problèmes pour trouver la route qui va de Nice à Paris. Toujours est-il que bien installé à 6 000 pieds, je tourne, je tourne, jusqu'à ce que mon radar d'approche me signale la piste. Alors commence la difficulté. Et oui, il faut descendre, oui, mais pas trop vite. Il faut garder une assiette correcte, réduire les gaz, mais pas trop, adopter une bonne vitesse, tenir le bon cap. Oui, ca y est presque, je vois la piste devant moi L'indicateur d'atterrissage IFR, les voyants 6 et 7 m'indiquent un peu plus haut, un peu plus à droite. Marrant le mec qui a programmé ce jeu, sorti de ma Ford Fiesta j'ai du mal à assimiler. Boom. L'écran devient rouge. Point n'est besoin de message pour comprendre que si j'avais cinquante personnes à mon bord, on aurai installé une chapelle ardente.

N'empêche que j'ai joué deux heures. Un conseil, achtetez ATI 42. Vous saurez une fois dans votre vie, ce que c'est que d'être pilote de ligne intérieur.

ATI 42 Editions Minipuce
Prix 200 Francs environ

INITIATION NIVEAU 1

Nous aborderons principalement les fichiers relatifs dans cette rubrique intiation. En effet, dans le dernier numéro, les fichiers séquentuellement avaient été traités. Attention, les fichiers relatifs qui sont beaucoup plus pratiques d'utilisation que les fichiers séquentiels ne sont disponibles que sur l'Exelmémoire et l'unité de disquettes

Généralités

Dans le dernier numéro d'EV, nous avions défini les fichiers comme des classeurs. Chaque feuille correspondait à un enregistrement, et la lecture et l'écriture dans ce fichier se faisaient séquentiellement, c'est-à-dire, que les enregistrements étaient écrits ou lus toujours dans le même ordre. Schématiquement, si vous voulez lire le 12 ème enregistrement (lire la douzième page du classeur), il faut lire les 11 premiers enregistrements, ou, en imageant un peu plus, si vous voulez ouvrir votre classeur directement à la 12 ème page, vous ne le pouvez pas, vous devez obligatoirement tourner les onze premières pages. A l'opposé, un fichier "RELATIF" peut être considéré comme un classeur équipé d'onglets pour repérer chaque page; et ainsi, on pourra ouvrir le classeur à la page désirée. Reprenons





l'exemple énoncé ci-dessus. Si vous désirez ouvrir votre classeur à la 12 ème page, vous n'aurez qu'à repérer la page en lisant les onglets et à ouvrir le classeur directement à l'endroit voulu.

Différence entre les fichiers séquentiels et les fichiers relatifs.

Il existe des différences notoires entre ces deux types de fichiers. Nous allons les analyser en fonction de leur apparition dans les procédures d'ouverture / d'écriture / de lecture et de fermeture d'un fichier. Mais avant tout, la grande différence entre un fichier séquentiel et un fichier relatif réside dans la façon d'accéder aux enregistrements. En effet, chaque enregistrement est repéré par un numéro.
En fonction de ce numéro d'enregistrement, beaucoup des procédures qui étaient obligatoires avec les fichiers séquentiels deviennent inutiles avec les fichiers relatifs. C'est notamment le cas lorsque l'utilisateur souhaite une mise à jour du fichier séquentiel. Nous verrons ce genre de manipulation dans un prochain numéro. Examinons une à une les différences d'écriture entre les fichiers séquentiels et les fichiers relatifs.

Ouverture d'un fichier relatif

OPEN #I, "100.nom de fichier", RELATIVE, mode d'ouverture

Tout comme pour les fichiers séquentiels, l'ouverture d'un fichier relatif ne diffère qu'avec l'ajout du terme **"RELATIVE"**. Pour les autres termes, rien de très différent. (cf EV 9). Dès que le **CROS** (système d'exploitation de l'Exelmémoire) ou le **DOS** (système d'exploitation du lecteur de disquettes) rencontre ce terme, ces derniers "comprennent" que le fichier sera une fichier relatif. Ceci entraînera une organisation physique du fichier différente d'une organisation séquentielle. Deux octets seront réservés pour écrire le numéro d'enregistrement avant l'enregistrement même.

I	enregistrement
I+1	enregistrement
I+2	enregistrement

Ecriture d'un fichier "RELATIF"

Une fois le fichier relatif ouvert, vous pouvez spécifier le numéro d'enregistrement. Ceci se réalise grâce à l'écriture du terme **REC** suivi d'un nombre qui peut être compris entre **0** et **32768** (2 à puissance 15). Si le terme REC est omis, le premier enregistrement portera le numéro zéro. Nous allons prendre l'exemple d'un fichier adresse pour un particulier.

Chaque client possède généralement un numéro d'enregistrement. En fonction du volume de clientèle de l'entreprise, ce numéro pourra être écrit sur 2 ou 3 chiffres. Chaque fiche client supportera: le nom, le prénom, l'adresse, le code postal, la ville et une rubrique divers. Grâce au fichier relatif, il sera possible de pointer sur la bonne fiche, uniquement en précisant le numéro d'enregistrement.

Le programme **BASIC 1.1** crée un fichier relatif en mode lecture / écriture. Mais ce dernier est plus particulièrement destiné à l'écriture des différentes données dans un enregistrement. Remarquez qu'à la ligne 140, lors de la procédure d'ouverture du fichier, on a précisé **"RELATIVE"**. Les lignes **150 à 200** constituent la saisie des informations. La ligne 210 demande le numéro d'enregistrement que l'on déclare comme la variable NE . A la ligne 230, on teste si la variable NE, autrement dit, le

```
100 '################################################
110 '#        FIC REL  ECRITURE                     #
120 '################################################
130 CLS
140 OPEN #1 "2,ADRESSE.FIC",RELATIVE,OUTPUT
150 CLS:LOCATE (8,3):PRINT "N ENREGISTREMENT : ';
160 ACCEPT BEEP VALIDATE(NUMERIC)SIZE(3)NULL(0) NE
170 IF NE=0 THEN 260
180 LOCATE (10,3):INPUT "NOM : ";N$
190 LOCATE (12,3):INPUT "PRENOM : ";P$
200 LOCATE (14,3):INPUT "ADRESSE : ";A$
210 LOCATE (16,3):INPUT "CODE POSTAL : ";CP$
220 LOCATE (18,3):INPUT "VILLE : ",V$
230 LOCATE (20,3):INPUT "DIVERS : ";D$
240 PRINT #1,REC NE,N$;",";P$;",";A$;",";CP$;" ";V$;",";D$
250 GOTO 150
260 CLOSE #1
```

Ligne 140 : Procédure d'ouverture

Lignes 150-160 : Introduction du numero d'enregistrement [3 chiffres maxi.] .Introduire la valeur 0 pour la fin du programme

Lignes 170 : Test sur la valeur du numero

Lignes 180 230 : Saisie des informations

Ligne 240 : Ecriture de l'enregistrement

Ligne 250 : Renvoi à la ligne 150

Ligne 260 : Fermeture du fichier

numéro d'enregistrement, est égale à 0. Si NE est égal à zéro, le fichier sera fermé. La ligne **240** écrit dans l'enregistrement spécifié par **NE**, les données concaténées: Le nom, le prénom, l'adresse, le code postal, la ville sont écrits dans le même enregistrement. La ligne 250 renvoie le programme à la ligne 150, ce qui permet de saisir une autre adresse. Pour arrêter la saisie, il suffit de ne pas taper de caractères lors de l'introduction du nom (ligne **170**) Tout comme pour les fichiers séquentiels, le fichier doit être fermé après avoir écrit le dernier enregistrement. (ligne 260).

Le programme BASIC 1.2 , destiné à lire le fichier créé par le programme 1.1, peut au premier abord, paraître plus compliqué que son homologue . En effet, nous trouvons dans le programme des instructions que, peut-être, vous n'avez pas l'habitude de rencontrer. Ces instructions particulières qui sont **ON WARNING** et **ON ERROR** seront discutés dans la partie initiation niveau 2 de la revue. Pour l'instant, nous ne nous préoccuperons pas de ces instructions chargées de gérer les erreurs. La ligne **170** permet d'ouvrir le fichier **ADRESSE.FIC** en lecture. A la ligne 190, le programme demande à l'utilisateur de donner le numéro d'enregistrement, **NE** . La prochaine étape consiste à tester si le numéro d'enregistrement NE est égal à zéro. Si c'est le cas, le fichier est fermé. La ligne 210 permet de lire l'enregistrement. Attention, n'oubliez pas que **ENR$** contiendra le nom, le **PRÉNOM**, l'adresse, le code postal et la ville.

La seule difficulté qui peut survenir dans la lecture d'un fichier relatif est la lecture d'un numéro d'enregistrement qui n'existe pas. Si c'est le cas, une erreur est générée. C'est justement dans le cas où cette situation se produit que l'on a écrit une gestion des erreurs.

Lecture d'un fichier "Relatif"

Un fichier relatif peut être lu de deux manières différentes: Soit en lecture séquentielle, soit en lecture "RELATIF". Nous allons lire de façon séquentielle le fichier que nous venons de créer. Pour cela, nous allons taper un petit programme (cf listing 1.3). La procédure d'ouverture du fichier est déclarée en mode séquentiel, le terme "RELATIF" n'est pas précisé. Dans ce cadre, la lecture du fichier est en tout point similaire à une lecture



```
100 '##################################
110 '#       FIC REL LECTURE          #
120 '##################################
130 CLS
140 ON ERROR 300
170 OPEN #1,"2.ADRESSE.FIC",RELATIVE,INPUT
180 CLS
185 LOCATE (10,1)
190 INPUT "N ENREGISTREMENT : ";NE
200 IF NE=0 THEN 290
210 INPUT #1,REC NE,N$,P$,A$,CP$,V$,D$
220 PRINT N$
230 PRINT P$
240 PRINT A$
250 PRINT CP$
260 PRINT V$
270 PRINT D$
275 ACCEPT BEEP VALIDATE("NUMERIC")SIZE(1)NULL(0),I
280 GOTO 180
290 CLOSE #1
295 END
300 CALL ERR(CODE,TYPE,FICH,LINE)
310 IF TYPE()7 THEN RETURN 290
315 PRINT "ENREGISTREMENT INEXISTANT"
317 PAUSE
320 ON ERROR 340
330 CLOSE #1
340 RETURN 130
```

Ligne 140 : Traitement des erreurs

Lignes 170 : Ouverture du fichier ADRESSE.FIC en mode lecture

Ligne 180 : Introduction du numero d'enregistrement

Ligne 200 : Test sur la valeur du numero. Tapez 0 pour fin

Ligne 210 : Lecture de l'enregistrement

Lignes 220-275 : Affichage des informations

Ligne 280 : Renvoi a la ligne 180

Ligne 290 : Fermeture du fichier

Lignes 300-340 : Gestion des erreurs

Ligne 310 : Si l'erreur ne provient pas d'un enregistrement inexistant, alors retour a la ligne 290

Ligne 320-340 : Fermeture du fichier dans le BASIC

séquentielle. Vous remarquerez que la lecture s'effectue sans avoir à préciser le numéro d'enregistrement. Toutefois, nous sommes dans l'obligation de tester la fin de fichier (ligne 150). La ligne 180 lit l'enregistrement **ENR$**. Une fois l'enregistrement lu, il est affiché. La ligne 200 permet d'arrêter la lecture des données jusqu'à ce que l'utilisateur appuye sur la touche [ENVOI]. Il est possible de lire un fichier "RELATIF" en mode séquentiel; par contre il est impossible de lire un fichier séquentiel en mode "RELATIF".

Utilisation de CALL RESTORE avec les fichiers "RELATIF"

La fonction **CALL RESTORE #LUNO, *variable*** peut être utilisée avec les fichiers relatifs comme avec les fichiers séquentiels. Toutefois, le résultat de la fonction CALL RESTORE est différent suivant que les fichiers sont écrits sur Exelmémoire ou sur disquettes. En effet, l'organisation des fichiers "RELATIF" sur l'Exelmémoire est différente de l'organisation sur disquettes. **CALL RESTORE** ne peut être utilisée qu'en lecture séquentielle.

Différence entre un fichier "RELATIF" sur Exelmémoire et sur disquettes

Nous avons vu que la grande différence entre les fichiers séquentiels et les fichiers "RELATIF" se situe au niveau de l'accès à un enregistrement. En effet, grâce au numéro d'enregistrement (REC), il est possible de pointer directement sur l'enregistrement souhaité. Dans le processus de création d'un fichier "RELATIF", si l'enregistrement n° 50 est écrit , les quarante neuf premiers enregistrements sont créés même s'il n'existaient pas



auparavant

Ceci est valable pour les fichiers "RELATIF" sur disquettes. Il en va autrement, faute de place, sur l'Exelmémoire. Sur cette dernière, le fait d'écrire le cinquantième enregistrement ne crée pas les quarante neuf premiers. Les enregistrements sont organisés de façon séquentielle. Rassurez-vous, même si les enregistrements sont rangés de façon séquentielle, on peut accéder à ces derniers par leur numéro d'enregistrement.
A la lumière de cette dernière explication, il faut être très prudent lorsque l'on utilise la fonction **CALL RESTORE**.
Si le fichier "RELATIF" est lu de manière séquentielle sur Exelmémoire, l'utilisation de CALL RESTORE permettra de pointer sur l'enregistrement précisé. Si par exemple, vous écrivez: CALL RESTORE #1, 2, vous pointerez sur le deuxième enregistrement du fichier. Ceci n'implique pas que ce soit l'enregistrement dont le numéro est 2.
Si le fichier "RELATIF" est sur disquette, l'utilisation du CALL RESTORE ne pose plus aucun problème et vous êtes sûr de pointer sur le numéro d'enregistrement spécifié dans CALL RESTORE.

Utilisation de CALL EOF

La fonction **CALL EOF #LUNO,** *variable* peut être utilisée aussi bien avec les fichiers "RELATIF" qu'avec les fichiers séquentiels. Tout comme pour les fichiers séquentiels, cette fonction retourne dans *variable* le nombre d'enregistrements restant à lire dans le fichier.

Quelques exemples de fichiers relatifs

Tous les fichiers séquentiels peuvent être adaptés en fichiers "RELATIF", néanmoins, on n'utilisera un fichies "RELATIF" que dans des cas bien précis, notamment si le fichier nécessite une mise à jour quasiment quotidienne. Nous allons vous donner un exemple d'application faisant intervenir un fichier "RELATIF" avec sa mise à jour. L'exemple que nous vous proposons est un programme qui gère le championat de première division de football. Les fichiers qui seront manipulés sont des fichiers de type RELATIVE.

L'application que nous vous proposons est un programme de gestion de championnat de football de première division.

Ce logiciel permet la gestion des meilleurs marqueurs du championnat.

Ce logiciel utilise l'Exelmémoire pour stocker les fichiers. Les fichiers sont de type RELATIVE

FOOT BALL

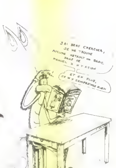
```
1 DIM R(20,2),T(20),V$(20)
10 CLS
30 LOCATE (1,16):PRINT 'FOOTBALL'
40 LOCATE (4,3):PRINT "Creation EQUIPES            [ 1 ]"
50 LOCATE (6,3):PRINT "Creation JOUEURS            [ 2 ]"
60 LOCATE (8,3):PRINT "Creation RENCONTRES         [ 3 ]"
70 LOCATE (10,3) PRINT "Creation BUTEURS           [ 4 ]"
80 LOCATE (12 3)·PRINT "Classement general         [ 5 ]"
90 LOCATE (14,3) PRINT "Classement buteurs         [ 6 ]"
100 LOCATE (16,3)·PRINT "Classement meilleur attaque [ 7 ]"
110 LOCATE (18,3)·PRINT "Classement meilleur defense [ 8 ]"
120 LOCATE (20,3)·PRINT "Fin                        [ 9 ]"
130 LOCATE (22,3)·ACCEPT BEEP VALIDATE(NUMERIC)SIZE(1)NULL(0),C
140 IF C=0 THEN 130
150 ON C GOTO 200,400,600,800,1000,1400,1800,2200,160
155 GOTO 10
160 CLS
170 ON ERROR 190
180 CLOSE #1:CLOSE #2:CLOSE #3:CLOSE #4
190 END
200 CLS
210 OPEN #1,"2.EQUIPES",RELATIVE,OUTPUT,VARIABLE 50
220 I=I+1
225 LOCATE (13 2):PRINT "Nom EQUIPE : "
227 LOCATE (13,15):ACCEPT BEEP VALIDATE(ALPHA)SIZE(25)NULL("x"),N$
228 V$(I)=N$
229 FOR J=1 TO I-1
230 IF V$(J)=N$THEN 225
231 NEXT J
235 IF N$="x"THEN 270
240 MG,MP MN,BM,BE,GO P=0
250 PRINT #1,REC I,N$;",";MG;",";MP;",";MN;",";BM;",";BE;",";GO;",";P
255 CLS
260 GOTO 220
270 CLOSE #1
280 GOTO 10
400 CLS
401 I=0
402 ON ERROR 10500
404 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
406 ON ERROR 10600
408 I=I+1
410 INPUT #2 REC I,N$,P$,AJ,BJ
412 GOTO 408
415 !
420 LOCATE (10,2)·PRINT "Nom : "
425 LOCATE (10,8)·ACCEPT BEEP VALIDATE(ALPHA)SIZE(20)NULL(" "),NJ$
427 IF NJ$=" "THEN 500
428 IF I=0 THEN 430
429 GOTO 10700
430 LOCATE (12,2)·PRINT "Prenom : "
435 LOCATE (12,11)·ACCEPT BEEP VALIDATE(ALPHA)SIZE(15)NULL("x");PJ$
437 IF PJ$="x"THEN 435
440 LOCATE (14 2):PRINT "Age : "
445 LOCATE (14,8):ACCEPT BEEP VALIDATE(NUMERIC)SIZE(2)NULL(0),AJ
447 IF AJ=0 THEN 445
448 IF AJ<15 OR AJ>50 THEN 445
449 IF I=0 THEN 10800
450 OPEN #2,"2.JOUEURS",RELATIVE,APPEND,VARIABLE 50
460 BJ=0
467 I=I+1
470 PRINT #2,REC I,NJ$;",";PJ$;",";AJ;",";BJ
475 CLS
477 CLOSE #2
480 GOTO 415
500 GOTO 10
600 CLS
620 LOCATE (10,2):PRINT "Nom Equipe : "
625 LOCATE (10,15):ACCEPT BEEP VALIDATE(ALPHA)SIZE(20)NULL(" "),N$
627 IF N$=" "THEN 700
628 GOTO 10200
630 LOCATE (12,2):PRINT "Nbr but marque : "
635 LOCATE (12,19)·ACCEPT BEEP VALIDATE(NUMERIC)SIZE(2)NULL(20),BMI
637 IF BMI=20 THEN 635
640 LOCATE (14,2):PRINT "Nbr but encaisse : "
645 LOCATE (14,21):ACCEPT BEEP VALIDATE(NUMERIC)SIZE(2)NULL(20),BEI
647 IF BEI=20 THEN 645
650 PRI=BMI-BEI:IF PRI>0 THEN PRI=2
652 IF PRI=0 THEN PRI=1
655 IF PRI<0 THEN PRI=0
656 ON ERROR 10650
657 OPEN #3,"2.RENCONTRE",APPEND,VARIABLE 40
660 PRINT #3,N$;",";BMI;",";BEI;",";PRI
662 CLOSE #3
665 CLS
670 GOTO 620
700 GOTO 10
800 CLS
810 OPEN #4,"2.BUTEURS",OUTPUT,VARIABLE 30
820 LOCATE (10,2):PRINT "Nom joueur : "
825 LOCATE (10,14):ACCEPT BEEP VALIDATE(ALPHA)SIZE(20)NULL(" "),NJ$
827 IF NJ$=" "THEN 870
840 LOCATE (12,2):PRINT "Nbr but marque : "
845 LOCATE (12,19):ACCEPT BEEP VALIDATE(NUMERIC)SIZE(2)NULL(0),BJI
847 IF BJI=0 THEN 845
850 PRINT #4,NJ$;",";BJI
860 GOTO 820
865 CLS
870 CLOSE #4
880 GOTO 10
1000 CLS
1007 ON ERROR 1700
1010 OPEN #3,"2.RENCONTRE",INPUT,VARIABLE 40
1012 CALL EOF#3,A
1015 IF A=0 THEN 1075
1020 INPUT #3 NJ$,BMI,BEI,PRI
1022 GOTO 10400
1025 INPUT #1,REC I,N$,MG,MP,MN,BM,BE,GO,P
1030 BM=BM+BMI
1035 BE=BE+BEI
1040 GO=BM-BE
1045 P=P+PRI
1050 IF PRI=0 THEN MP=MP+1
1055 IF PRI=1 THEN MN=MN+1
1060 IF PRI=2 THEN MG=MG+1
1065 PRINT #1,REC I,N$;",";MG;",";MP;",";MN;",";BM;",";BE;",";GO;",";P
1067 CLOSE #1
```

```
1070 GOTO 1012
1075 CLOSE #3
1080 DELETE "2.RENCONTRE"
1085 NE=0
1090 OPEN #1,"2.EQUIPES",INPUT,VARIABLE 50,RELATIVE
1095 ON ERROR 3000
1105 NE=NE+1
1110 INPUT #1,REC NE N$,MG,MP,MN,BM,BE,GO,P
1115 GOTO 1105
1123 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
1125 FOR I=1 TO NE
1130 INPUT #1,REC I,N$,MG,MP,MN,BM,BE,GO,P
1135 R(I,1)=P:R(I,2)=GO
1140 T(I)=I
1142 P,GO=0
1145 NEXT I
1147 CLOSE #1
1150 FOR I=1 TO NE-1
1155 IF R(T(I),1)>R(T(I+1),1)THEN 1175
1160 IF R(T(I),1)<>R(T(I+1),1)THEN 1170
1165 IF R(T(I),2)>=R(T(I+1),2)THEN 1175
1170 X=T(I):T(I)=T(I+1):T(I+1)=X
1175 NEXT I
1180 FOR J=1 TO NE 1
1185 IF R(T(J),1)<R(T(J+1),1)THEN 1150
1187 IF R(T(J),1)<=R(T(J+1),1)THEN IF R(T(J),2)<R(T(J+1),2)THEN 1150
1190 NEXT J
1194 CLS
1195 LOCATE (10,4):PRINT "Edition sur imprimante [O/N] : "
1196 LOCATE (10,35):ACCEPT BEEP VALIDATE("ON")SIZE(1),M$
1197 IF M$="O"THEN 7000
1200 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
1205 CLS
1230 FOR I=1 TO NE
1235 INPUT #1,REC T(I),N$,MG,MF,MN,BM,BE,GO,P
1240 LOCATE (6+I,1):PRINT I
1245 LOCATE (6+I,4):PRINT N$
1250 LOCATE (6+I,19):PRINT MG
1255 LOCATE (6+I,22):PRINT MF
1260 LOCATE (6+I,25):PRINT MN
1265 LOCATE (6+I,28):PRINT BM
1270 LOCATE (6+I,32):PRINT BE
1275 LOCATE (6+I,35):PRINT GO
1280 LOCATE (6+I,38):PRINT P
1282 NEXT I
1285 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
1290 CLOSE #1
1295 GOTO 10
1400 CLS
1405 OPEN #2,"2.JOUEURS",RELATIVE,UPDATE,VARIABLE 50
1407 ON ERROR 1650
1410 OPEN #4,"2.BUTEURS",INPUT,VARIABLE 30
1415 CALL EOF#4,A
1417 IF A=0 THEN 1455
1418 C=0
1420 INPUT #4,MJI$,BJI
1425 C=C+1
1430 INPUT #2,REC C,MJ$,PJ$,AJ,BJ
1435 IF MJI$<>MJ$THEN 1425
1440 BJ=BJ+BJI
1445 PRINT #2,REC C,MJ$;",";PJ$;",";AJ;",";BJ
1447 MJI$="":NE,BJI=0
1450 GOTO 1415
1455 CLOSE #2:CLOSE #4
1460 DELETE "2.BUTEURS"
1465 C=0
1470 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
1475 ON ERROR 4000
1480 C=C+1
1485 INPUT #2,REC C,MJ$,PJ$,AJ,BJ
1490 GOTO 1480
1500 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
1505 FOR I=1 TO C
1510 INPUT #2,REC I,MJ$,PJ$,AJ,BJ
1515 R(I,1)=BJ:T(I)=I
1520 BJ=0
1525 NEXT I
1530 CLOSE #2
1535 FOR I=1 TO C-1
1540 IF R(T(I),1)>=R(T(I+1),1)THEN 1550
1545 X=T(I):T(I)=T(I+1):T(I+1)=X
1550 NEXT I
1555 FOR J=1 TO C 1
1560 IF R(T(J),1)<R(T(J+1),1)THEN 1535
1565 NEXT J
1566 CLS
1567 LOCATE (10,4):PRINT "Edition sur imprimante [O/N] : "
1568 LOCATE (10,35):ACCEPT BEEP VALIDATE("ON")SIZE(1),M$
1569 IF M$="O"THEN 8000
1570 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
1575 CLS
1580 FOR I=1 TO C
1585 INPUT #2,REC T(I),NJ$,PJ$,AJ,BJ
1587 IF BJ=0 THEN 1620
1590 LOCATE (6+I,1):PRINT NJ$
1595 LOCATE (6+I,18):PRINT PJ$
1600 LOCATE (6+I,30):PRINT AJ
1610 LOCATE (6+I,33):PRINT BJ
1615 NEXT I
1620 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
1625 CLOSE #2
1630 GOTO 10
1650 CALL ERR(CODE,TYPE,FICH,LINE)
1655 IF LINE<>1410 THEN RETURN 160
1670 CLS
1675 LOCATE (10,5):PRINT "Fichier BUTEURS non cree"
1677 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
1680 ON ERROR 1690
1685 CLOSE #2:CLOSE #4
1690 RETURN 10
1700 CALL ERR(CODE,TYPE,FICH,LINE)
1705 IF LINE<>1010 THEN RETURN 160
1710 CLS
1715 LOCATE (10,1):PRINT "Fichier RENCONTRE non cree "
1720 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
1725 ON ERROR 1735
1730 CLOSE #3
1735 RETURN 10
1800 CLS
1805 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
1810 ON ERROR 5000
1815 NE=0
1820 NE=NE+1
1825 INPUT #1,REC NE,N$,MG,MF,MN,BM,BE,GO,P
1830 GOTO 1820
1835 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
1840 FOR I=1 TO NE
1845 INPUT #1,REC I,N$,MG,MF,MN,BM,BE,GO,P
1850 R(I,1)=BM:R(I,2)=GO:T(I)=I
1855 BM,GO=0
1860 NEXT I
1865 CLOSE #1
1870 FOR I=1 TO NE-1
1875 IF R(T(I),1)>R(T(I+1),1)THEN 1895
1880 IF R(T(I),1)<>R(T(I+1),1)THEN 1890
1885 IF R(T(I),2)>=R(T(I+1),2)THEN 1895
1890 X=T(I):T(I)=T(I+1):T(I+1)=X
1895 NEXT I
1900 FOR J=1 TO NE-1
1905 IF R(T(J),1)<R(T(J+1),1)THEN 1870
1907 IF R(T(J),1)<R(T(J+1),1)THEN IF R(T(J),2)<=R(T(J+1),2)THEN 1870
1910 NEXT J
1911 CLS
1912 LOCATE (10,4):PRINT "Edition sur imprimante [O/N] : "
1913 LOCATE (10,35):ACCEPT BEEP VALIDATE("ON")SIZE(1),M$
1914 IF M$="O"THEN 9000
1915 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
1920 CLS
1925 FOR I=1 TO NE
1930 INPUT #1,REC T(I),N$,MG,MP,MN,BM,BE,GO,P
1935 LOCATE (6+I,1):PRINT I
1940 LOCATE (6+I,4):PRINT N$
1945 LOCATE (6+I,28):PRINT BM
1950 LOCATE (6+I,32):PRINT GO
1955 NEXT I
1960 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
1965 CLOSE #1
1970 GOTO 10
2200 CLS
2206 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
2210 ON ERROR 6000
2215 NE=0
2220 NE=NE+1
2225 INPUT #1,REC NE,N$,MG,MF,MN,BM,BE,GO,P
2230 GOTO 2220
2235 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
2240 FOR I=1 TO NE
2245 INPUT #1,REC I,N$,MG,MF,MN,BM,BE,GO,P
2250 R(I,1)=BE:R(I,2)=GO:T(I)=I
2255 BE,GO=0
2260 NEXT I
2265 CLOSE #1
2270 FOR I=1 TO NE-1
2275 IF R(T(I),1)<R(T(I+1),1)THEN 2295
2280 IF R(T(I),1)<>R(T(I+1),1)THEN 2290
2285 IF R(T(I),2)>=R(T(I+1),2)THEN 2295
2290 X=T(I):T(I)=T(I+1):T(I+1)=X
2295 NEXT I
2300 FOR J=1 TO NE-1
2305 IF R(T(J),1)>R(T(J+1),1)THEN 2270
2307 IF R(T(J),1)=R(T(J+1),1)THEN IF R(T(J),2)<R(T(J+1),2)THEN 2270
2310 NEXT J
2315 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
2316 CLS
2317 LOCATE (10,4):PRINT "Edition sur imprimante [O/N] : "
2318 LOCATE (10,35):ACCEPT BEEP VALIDATE("ON")SIZE(1),M$
2319 IF M$="O"THEN 10000
2320 CLS
2325 FOR I=1 TO NE
2330 INPUT #1,REC T(I),N$,MG,NF,MN,BM,BE,GO,P
2335 LOCATE (6+I,1):PRINT I
2340 LOCATE (6+I,4):PRINT N$
2345 LOCATE (6+I,28):PRINT BE
2350 LOCATE (6+I,32):PRINT GO
2355 NEXT I
2360 ACCEPT BEEP VALIDATE(" ")SIZE(1),M$
2365 CLOSE #1
2370 GOTO 10
3000 CALL ERR(CODE,TYPE,FICH,LINE)
3005 IF LINE<>1110 THEN RETURN 160
3010 NE=NE-1
3012 ON ERROR 3015
3013 CLOSE #1
3015 RETURN 1123
```

```
4000 CALL ERR(CODE,TYPE,FICH,LINE)
4005 IF LINE()1485 THEN RETURN 160
4010 C=C-1
4015 ON ERROR 4025
4020 CLOSE #2
4025 RETURN 1500
5000 CALL ERR(CODE,TYPE,FICH,LINE)
5005 IF LINE()1825 THEN RETURN 160
5010 NE=NE-1
5015 ON ERROR 5025
5020 CLOSE #1
5025 RETURN 1835
6000 CALL ERR(CODE,TYPE,FICH,LINE)
6005 IF LINE()2225 THEN RETURN 160
6010 NE=NE-1
6015 ON ERROR 6025
6020 CLOSE #1
6025 RETURN 2235
7000 CLS
7005 OPEN #5,"20",OUTPUT,VARIABLE 80
7010 A$="*":B$=RPT$("*",66)
7015 PRINT #5,TAB(24);"CLASSEMENT GENERAL"
7025 FOR I=1 TO 10
7030 PRINT #5
7035 NEXT I
7040 PRINT #5,B$
7045 C$="*CLAS*          NOM           * MG * MP * MN * BN * BE * GO * PT *"
7050 PRINT #5,C$
7055 PRINT #5,B$
7057 PRINT #5
7060 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
7065 FOR I=1 TO NE
7070 INPUT #1,REC T(I),N$,MG,MP,MN,BM,BE,GO,P
7075 PRINT #5,TAB(3);I;TAB(8);N$;TAB(33);MG;TAB(38);MP;TAB(43);MN;
7080 PRINT #5,TAB(48);BM;TAB(53);BE;TAB(58);GO;TAB(63);P
7085 PRINT #5
7087 NEXT I
7089 PRINT #5,B$
7090 CLOSE #5:CLOSE #1
7095 GOTO 10
8000 CLS
8005 OPEN #5,"20",OUTPUT,VARIABLE 80
8010 A$="*":B$=RPT$("*",66)
8015 PRINT #5,TAB(26);"CLASSEMENT BUTEURS'
8025 FOR I=1 TO 10
8030 PRINT #5
8035 NEXT I
8040 PRINT #5,B$
8045 C$="*CLAS*          NOM           *      PRENOM      * AGE * Nbr but *"
8050 PRINT #5,C$
8055 PRINT #5,B$
8060 PRINT #5
8065 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
8070 FOR I=1 TO C
8075 INPUT #2,REC T(I),NJ$,PJ$,AJ,BJ
8077 IF BJ=0 THEN 8092
8080 PRINT #5,TAB(3);I;TAB(8);NJ$;TAB(33);PJ$;TAB(49);AJ;TAB(50),BJ
8085 PRINT #5
8090 NEXT I
8092 PRINT #5
8095 PRINT #5,B$
8096 CLOSE #2:CLOSE #5
8097 GOTO 10
9000 CLS
9005 OPEN #5,"20",OUTPUT,VARIABLE 80
9010 A$='*":B$=RPT$('*",43)
9015 PRINT #5,TAB(3);"CLASSEMENT / ATTAQUE"
9020 FOR I=1 TO 10
9025 PRINT #5
9030 NEXT I
9035 PRINT #5,B$
9040 C$="*CLAS*         EQUIPE         * BUTS * GO *"
9045 PRINT #5,C$
9050 PRINT #5,B$
9055 PRINT #5
9060 OPEN #1."2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
9065 FOR I=1 TO NE
9070 INPUT #1,REC T(I),N$,MG,MP,MN,BM,BE,GO,P
9075 PRINT #5,TAB(3);I;TAB(8);N$;TAB(33);BM;TAB(39);GO
9077 PRINT #5
9080 NEXT I
9082 PRINT #5 B$
9085 CLOSE #1.CLOSE #5
9090 GOTO 10
```

```
10000 CLS
10005 OPEN #5,"20",OUTPUT,VARIABLE 80
10010 A$="*":B$=RPT$("*",43)
10015 PRINT #5,TAB(12);"CLASSEMENT / DEFENSE"
10020 FOR I=1 TO 10
10025 PRINT #5
10030 NEXT I
10035 PRINT #5,B$
10040 C$="*CLAS*         EQUIPE         * BUTS * GO *"
10045 PRINT #5,C$
10050 PRINT #5,B$
10055 PRINT #5
10057 CLOSE #1
10060 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
10065 FOR I=1 TO NE
10070 INPUT #1,REC T(I),N$,MG,NF,MN,BM,BE,GO,P
10075 PRINT #5,TAB(3);I;TAB(8);N$;TAB(33);BE;TAB(38);GO
10080 PRINT #5
10085 NEXT I
10090 PRINT #5,B$
10095 CLOSE #1:CLOSE #5
10100 GOTO 10
10200 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
10205 I=0
10210 ON ERROR 10300
10215 I=I+1
10220 INPUT #1,REC I,NJ$,MG,NE,MN,BM,BE,GO,F
10225 GOTO 10215
10230 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
10235 FOR NE=1 TO I
10240 INPUT #1,REC NE,NJ$,MG,NF,MN,BM,BE,GO,P
10245 IF NJ$=N$THEN 10280
10250 NEXT NE
10255 CLS
10260 LOCATE (10,2):PRINT "NOM EQUIPE INCORECTE "
10265 ACCEPT BEEP VALIDATE(" ")SIZE(1),W$
10270 CLOSE #1
10275 GOTO 620
10280 CLOSE #1
10285 GOTO 630
10300 CALL ERR(CODE,TYPE,FICH,LINE)
10305 IF LINE()10220 THEN RETURN 160
10310 I=I-1
10315 ON ERROR 10325
10320 CLOSE #1
10325 GOTO 10230
10400 OPEN #1,"2.EQUIPES",RELATIVE,INPUT,VARIABLE 50
10405 I=0
10410 I=I+1
10415 INPUT #1,REC I,N$,MG,MP,MN,BM,BE,GO,P
10420 IF N$=NJ$THEN 10430
10425 GOTO 10410
10430 CLOSE #1
10435 OPEN #1,"2.EQUIPES",RELATIVE,UPDATE,VARIABLE 50
10440 GOTO 1025
10500 CALL ERR(CODE,TYPE,FICH,LINE)
10505 IF LINE()404 THEN RETURN 160
10510 ON ERROR 10520
10515 CLOSE #2
10520 RETURN 415
10600 CALL ERR(CODE,TYPE,FICH,LINE)
10605 IF LINE()410 THEN RETURN 160
10610 I=I-1
10615 ON ERROR 10625
10620 CLOSE #2
10625 RETURN 415
10700 OPEN #2,"2.JOUEURS",RELATIVE,INPUT,VARIABLE 50
10702 FOR J=1 TO I
10705 INPUT #2,REC J,N$,P$,AJ,BJ
10710 IF N$=NJ$THEN 10730
10715 NEXT J
10720 CLOSE #2
10725 GOTO 430
10730 CLOSE #2
10735 GOTO 420
10800 OPEN #2,"2.JOUEURS",RELATIVE,OUTPUT,VARIABLE 50
10805 GOTO 460
10850 CALL ERR(CODE,TYPE,FICH,LINE)
10855 IF LINE()604 THEN RETURN 160
10860 ON ERROR 10870
10865 CLOSE #3
10870 OPEN #3,"2.RENCONTRE",OUTPUT,VARIABLE 40
10875 RETURN 620
```

INITIATION NIVEAU 2



L'EXLBASIC comporte un certain nombre d'instructions qui ne sont pratiquement jamais commentées dans les manuels d'initiation. Les instructions ON ERROR, ON WARNING, et le sous-programme intégré ERR permettent de lancer l'exécution de certaines actions lorsqu'une ou des erreurs sont détectées dans le déroulement d'un programme. Si le warning n'évoque pour vous que les feux de détresse de votre véhicule, nous venons à votre secours. Une erreur provoque l'arrêt de l'exécution d'un programme et un message d'erreur est affiché à l'écran. Le warning ne provoque pas l'arrêt du programme, mais néanmoins, un message de warning est affiché à l'écran. WARNING signifie avertissement.

LES WARNINGS

Il est possible, grâce à des instructions ON ERROR, et ON WARNING combinées au sous-programme ERR, de gérer les erreurs générées au cours d'un programme. Nous allons taper petit programme.

```
10 INPUT "TAPEZ UN NOM:";M$
20 L$=RPT$ (M$,100)
30 PRINT L$
```

Lancez ce programme en tapant RUN suivi de [ENVOI]. Si un nom de plus de trois lettres est fourni, l'exécution de la ligne 20 génèrera un WARNING car la chaîne L$ comporte plus de 255 caractères. Un message Err 36 Line 20 est affiché, mais le programme affiche néanmoins la chaîne L$ tronquée au 256 ème caractère. Le WARNING n'a pas interrompu le programme. Il est quand même dommage que dans un programme, un tel message soit affiché. L'instruction ON WARNING peut être utilisée avec plusieurs options.

ON WARNING NEXT

Utilisée avec l'option NEXT, l'instruction ON WARNING ne tiendra pas compte de l'avertissement, et aucun message d'erreur ne sera retourné.

```
5 ON WARNING NEXT
10 INPUT "TAPEZ UN NOM:";M$
20 L$=RPT$ (M$,100)
30 PRINT L$
40 GOTO 10
```

Si vous exécutez ce programme, aucun message d'erreur n'est affiché. Le programme se déroule normalement comme si aucun warning n'était intervenu. Utilisée de cette manière l'instruction ON WARNING ne tiendra aucun compte des warnings éventuels.

ON WARNING STOP

C'est le mode d'utilisation normal de l'instruction ON WARNING. Un message d'erreur est affiché, mais le programme continue son exécution. Lorsque l'EXL 100 est initialisé, ce mode de fonctionnement est prioritaire.

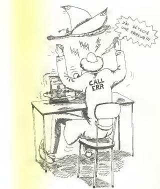
ON WARNING ERROR

Cette option permet de traiter des messages WARNINGS comme des messages d'erreur. De cette manière, les warnings pourront être gérés dans un sous-programme de traitement des erreurs.

LES ERREURS

Une erreur provoque l'arrêt du programme. Un message est alors affiché à l'écran. Lorsqu'une erreur survient dans un programme, vous pouvez décider de l'action qui sera effectuée en fonction de l'instruction ON ERROR. Plusieurs options sont disponibles avec cette instruction.

ON ERROR STOP



C'est le mode de fonctionnement normal de l'instruction ON ERROR. C'est-à-dire que lorsque l'EXL 100 est initialisé ou mis en marche, ce mode de détection des erreurs est prioritaire. Lorsqu'une erreur est générée, le programme est arrêté, et un message d'erreur est affiché à l'écran.

Un message d'erreur est matérialisé à l'écran par

Err (code) Line (numéro de ligne)

ON ERROR numéro de ligne

Utilisée de cette manière, l'instruction ON ERROR tranfère le contrôle du programme vers un sous-programme de gestion des erreurs situé au numéro de ligne précisé après le terme ERROR. Lorsqu'une erreur est générée dans le programme, le contrôle est tranféré au sous-programme. Le sous-programme de gestion des erreurs peut faire appel au sous-programme intégré CALL ERR

```
100 ON ERROR 1000
110 INPUT"DONNEZ UN NOMBRE:";N
120 R=SQR(R)
130 PRINT SQR
140 GOTO 100
1000 REM GESTION DES ERREURS
1010 CALL ERR(CODE,TYPE,FICH,LINE)
1020 PRINT "CODE=";CODE;"
";"TYPE=";TYPE;"          ";"FICHIER=".FIC;"
";"LIGNE:";LIGNE
1030 RETURN NEXT
RUN [ENVOI]
```

Ce programme demande l'introduction d'un nombre, et calcule ensuite la racine carrée de ce nombre. Si un nombre négatif est fourni, une erreur est générée. La ligne 100 transfère le contrôle du programme au sous-programme chargé de gérer les erreurs lorsqu'une erreur est détectée Le sous-programme à la ligne 1000 utilise le sous-programme intégré CALL ERR Ce sous-programme affecte à:

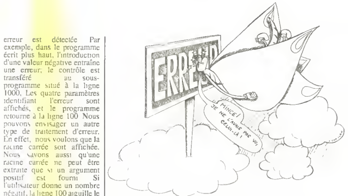
CODE, le code de l'erreur
TYPE, le type de l'erreur
FICH, le numéro logique de fichier concerné par l'erreur
LINE, le numéro de ligne de l'erreur détectée.

Ainsi, lorsqu'une erreur est détectée, le sous-programme CALL ERR permet d'indentifier à l'aide de quatre paramètres la nature d'une erreur.

Traitement des erreurs

Qu'appelle-t-on traitement des erreurs ? Le traitement des erreurs consiste à déterminer l'action la plus propice qui sera exécutée lorsqu'une





erreur est détectée Par exemple, dans le programme écrit plus haut, l'introduction d'une valeur négative entraîne une erreur; le contrôle est transféré au sous-programme situé à la ligne 1000, Les quatre paramètres identifiant l'erreur sont affichés, et le programme retourne à la ligne 100 Nous pouvons envisager un autre type de traitement d'erreur. En effet, nous voulons que la racine carrée soit affichée. Nous savons aussi qu'une racine carrée ne peut être extraite que si un argument positif est fourni Si l'utilisateur donne un nombre négatif, la ligne 100 aiguille le programme vers la gestion des erreurs Lorsque l'on essaie d'extraire une racine en fournisssant un nombre négatif, le code d'erreur 29 est retourné. Nous pouvons, en fonction de la valeur de ce code, décider d'une modification dans le traitement des erreurs. Remplaçons le ligne 1020 comme suit.

```
1020 IF CODE = 29
THEN N=ABS(N) :
RETURN 130
```

Le programme devient alors.

```
100 CLS"WBC":CALLCOLOR ("0WB")
110 ON ERROR 1000
120 INPUT"DONNEZ UN NOMBRE:"
;N
130 R=SQR(N)
140 ON ERROR 1000
150 PRINT R
160 GOTO 120
1000 CALL ERR
(CODE,TYPE,FICH,LIN)
1010 IF CODE = 29 THEN N=ABS(N):
RETURN 130
1020 PAUSE
1030 RETURN 110
```

Si le code de l'erreur détectée est égal à 29, la ligne 1020 prend la valeur absolue du nombre introduit. De cette manière, l'argument de la racine carrée sera toujours positif, et ainsi, la racine carrée pourra être affichée

Un sous-programme de gestion des erreurs se termine toujours par l'instruction RETURN. L'instruction RETURN peut être exploitée de différentes manières En effet, l'instruction RETURN lorsqu'elle est exécutée retourne le contrôle du programme à, ou aux lignes déterminées par le choix de l'option.

RETURN NEXT: Une fois le sous-programme de traitement des erreurs exécuté, le contrôle est transféré à la prochaine instruction suivant la source de l'erreur.

RETURN ***numéro de ligne*** Le contrôle est tranféré à la ligne précisée dans *numéligne de*

Dans le programme ci-dessus, uniquement la deuxième option est utilisée.

Ordre d'exécution des instructions ON ERROR ou ON WARNING

En fonction des options choisies après l'instruction RETURN qui matérialise la fin d'un sous-programme de gestion des erreurs, ces dernières pourront être totalement traitées ou non. En effet, lorsqu'une instruction ON ERROR est exécutée, son activité n'est que relative, et si le programme n'exécute pas cette instruction, celle-ci est inactive, et la gestion des erreurs ne sera pas effectuée. Autrement dit, regardez le programme d'exemple. Simulons un passage.

Lorsque le programme est exécuté pour la première fois, dès qu'une erreur est détectée, le contrôle est tranféré à la ligne 1000. Cette même ligne 1000 interprète le code d'erreur. La ligne 1010 teste le code de l'erreur. Si le code correspond bien à la valeur 29, la valeur absolue du nombre introduit est affectée à la variable N, et le contrôle est retournée à la ligne 130. **Supprimez la ligne 140**. Introduisez par deux fois des nombres négatifs. Lorsque la première valeur négative est fournie, le sous-programme de traitement des erreurs joue parfaitement son rôle.

DONNEZ UN NOMBRE
-81 [ENVOI]
9

DONNEZ UN NOMBRE
: -100
Err 29 Line 130

Si la ligne 140 est supprimée et qu'une valeur négative est fournie pour la deuxième fois, le programme est interrompu, et un message d'erreur est affiché à l'écran. En effet, une instruction ON ERROR n'est active qu'une seule fois. Pour que le sous-programme de gestion des erreurs fonctionne parfaitement, les instructions ON ERROR doivent être en permanence exécutées. La ligne 140 avait était écrite dans ce but. En effet, une fois que la racine carrée est calculée, l'instruction ON ERROR transfère le contrôle de l'exécution au sous-programme de traitement des erreurs. Le programme "repasse" toutes les fois sur la ligne 140.

Comme nous venons de le voir, il est possible de déclencher des actions en fonction des codes ou du type d'erreur. Nous vous conseillons de toujours prévoir un programme de traitement des erreurs. En effet, il n'y a rien de plus désagréable que d'utiliser un programme qui "plante" à chaque fois qu'une erreur est générée. Nous essayerons dans la limite du possible, d'établir tous nos programmes d'exemples avec un traitement d'erreur intégré

PAGE BRICOLAGE

Réalisez des extensions pour votre EXL 100

L'EXL 100, contrairement à la plupart des ordinateurs familiaux, possède une caractéristique très intéressante il est très facile d'augmenter ses possibilités grâce à ses deux emplacements prévus pour recevoir des cartes d'extension Exelvision vous propose déjà actuellement une vaste gamme de cartes d'extension: La carte imprimante parallèle et série, Exelmodem, Exeldrums, Exelmémoire, horloge temps réel, et décodeur ANTIOPE. Mais les applications possibles de l'EXL 100 sont illimitées, et certains utilisateurs peuvent avoir besoin d'interfaces très spécialisées. On peut imaginer des cartes d'interfaces analogiques, des cartes spécialisées pour les radioamateurs (CW, RTTY, SSTV), des cartes de commandes de robots, de radiateurs électriques, des cartes de synthèse musicale, etc..

Notre but est de vous expliquer de façon détaillée la manière de relier vos extensions sur les connecteurs situés à l'arrière de votre EXL 100 Nous verrons pour commencer, le connecteur d'extension d'Entrée / Sortie, le plus grand des deux (celui de gauche)

1. Brochage du connecteur et liste des signaux

Ce connecteur comporte 2 x 25 broches au pas de 2,54 mm Il est particulièrement prévu pour réaliser des extensions d'entrée / sortie, par opposition à des extensions de mémoire En effet, le bus d'adresses ne sort pas sur ce connecteur, mais deux signaux de validation correspondant à deux adresses sont présents En parcourant la liste des signaux (fig 1), on reconnaît le bus des données (DB7 à DB0), des signaux de contrôle (R / W, CLOCKOUT, ENABLE, ALATCH), et des alimentations (+ 5V, -5V, +12V, -12V) La présence de ces alimentations symétriques sera très appréciée par les réalisateurs d'interfaces analogiques, puisqu'elles permettront d'alimenter directement des amplificateurs opérationnels Les broches B1 à B25 sont situées du côté des composants. Les broches A25 et B25 sont les plus proches des connecteurs cassette et péritélévision.

2. Les signaux principaux

Ce sont le bus de données, et les signaux de contrôle:

PEREN1 : Peripheral Enable1

Passe à 0 lorsque l'adresse présente sur le bus d'adresse est >131



du bus de données lorsque le signal de validation est présent et la conservent jusqu'à l'inscription d'une autre valeur.

Puisqu'il y a deux signaux de validation disponibles sur le connecteur de l'EXL 100, nous allons utiliser l'un pour le port de sorties, l'autre pour le port d'entrées. Ces signaux de validation doivent être combinés avec ENABLE et R/W pour commander directement les ports.
Grâce à la ligne RSTB, le port de sorties est mis à zéro à la mise sous tension, ou lorsqu'on appuie sur la touche RESET.

PEREN2 Peripheral Enable 2

Passe à 0 lorsque l'adresse présente sur le bus d'adresse est > 132

RSTB: Reset

Passe à 0 à la mise sous tension, ou lorsqu'on appuie sur le bouton RESET

CLKOUT Horloge du TMS 7020 (voir figure 2)

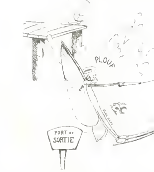
ALATCH Indique que le bus de données du TMS 7020 contient la partie basse de l'adresse.Le démultiplexage étant effectué sur l'EXL 100, le bus de données arrivant sur le connecteur ne contient plus l'adresse basse pendant que ALATCH est à 1

ENABLE Ce signal sert à valider les transferts à travers le bus de données (voir figure 2)

R / $\overline{W}$: READ / WRITE

Indique le sens de tranfert des données, vu du côté du microprocesseur

PERINT
Peripheral Interrupt
Entrée d'interruption du TMS 7020



3 Exemple : Cartes d'entrées / sorties parallèles

Pour illustrer l'utilisation des signaux décrits ci-dessus, réalisons le schéma d'une carte d'interface comprenant huit entrées et huit sorties (voir figure 3).
Le port d'entrées est constitué par un étage tampon à trois états (Tristate buffer) qui permet de n'envoyer les signaux externes sur le bus des données que lorsque le signal de validation est présent.
Le port des sorties est constitué par des bascules (D), qui prennent la valeur

L'utilisation d'une telle carte est extrêmement simple

En BASIC:

CALL POKE (305, A)

Envoi d'une valeur sur le port de sorties

CALL PEEK (306,A)

Lecture d'une valeur sur le port d'entrées

En assembleur:

82 31 MOVP A,P49

Envoi d'une valeur sur le port de sorties

80 32 MOVP P50,A

Lecture d'une valeur sur le port d'entrées

4. Interfaçage avec le monde extérieur

Le schéma décrit ci-dessus n'est utilisable que pour communiquer avec d'autres équipements électroniques qui fonctionnent avec des niveaux logiques TTL (0 et 5 Volts). Mais, pour commander des puissances plus importantes, on peut être obligé d'utiliser des relais. Si l'on veut isoler l'EXL 100 de l'équipement, on peut aussi utiliser des photocoupleurs.

4.1 Utilisation des relais

La figure 4 montre comment une sortie du 74LS244 peut commander un relais. Le transistor permet de faire passer, ou de couper le courant qui alimente la bobine du relais. Lorsque la sortie du 74LS244 est à zéro, le transistor est bloqué, et le relais ne colle pas

4.2 Utilisation de photocoupleurs

Si le courant à commuter est plus faible, on peut remplacer le relais par un photocoupleur (TIL 111 par exemple) Un photocoupleur est constitué d'une LED et d'un phototransistor. Il suffit de faire passer un courant dans la LED pour éclairer le phototransistor qui devient conducteur Avec le schéma de la figure 5, le transistor est bloqué lorsque la sortie du 74LS244 est à 1

5 Les entrées sorties "Série"

Cinq signaux se rapportent à ce mode de transmission:

RXD
Received DATA (Réception de données)

TXD
Transmitted DATA(Emission de données)

DTR
Data Terminal (Terminal de données prêt)

DSR
Data Set Ready (Porte de données prête)

SCLK
Serial Clock (Horloge série)

L'initialisation de tous les paramètres de la transmission peut être effectuée en BASIC par la commande OPEN.

10 OPEN #10, "10.B=a,N=b,S=c,P=d,T=e"

où a,b,c,d,e représentent des valeurs entières, et B, N, S, P, T représentent.

B permet de choisir la vitesse de transmission selon les égalités suivantes.

B	Vitesse en Bauds	Vitesse en caractères par sec
0	110	10
1	150	15
2	300	30
3	600	60
4	900	90
5	1200	120
6	2400	240
7	4800	480
8	9600	960
9	19200	1920

N permet de sélectionner le nombre de bits par caractère:
-7 pour 7 bits par caractère (valeur prise par défaut)
-8 pour 8 bits par caractère
S permet de sélectionner le nombre de bits de stop
-1 pour 1 bit de stop (valeur prise par défaut)
-2 pour 2 bits de stop
P permet de sélectionner la parité
-2 pour parité paire
-1 pour parité impaire
-0 pour absence de parité
T permet de sélectionner le protocole de fonctionnement des lignes DTR et DSR.
-0 DSR est mis au +12 Volts, la réponse doit être DTR au +12 Volts
-1 DSR est mis au -12 Volts, la réponse doit être DTR au +12 Volts
-2 DSR est mis au +12 Volts, la réponse doit être DTR au -12 Volts
-3 DSR est mis au -12 Volts, la réponse doit être DTR au -12 Volts
Ainsi, pour configurer la liaison pour 300 Bauds, 7 bits par caractère, 1 bit de stop, parité paire est DSR au + 12 Volts, on devra écrire

OPEN #10,"10.B=2,N=7,S=1, P=2,T=0"

L'émission des données se réalise à l'aide de l'instruction PRINT # du BASIC.

SCLK est l'horloge utilisée pour la transmission série. Sa fréquence est égale à 16 fois le débit binaire (19200 Hz pour 1200 Bauds). SCLK est habituellement une sortie, mais peut être aussi une entrée, si on utilise une horloge externe.

6. Les signaux vidéo et audio

Si vous utilisez un EXELMODEM, ou une carte d'interface imprimante, vous avez sûrement déjà essayé le mode incrustation du BASIC. Il permet d'afficher un texte sur une image TV. Tous les signaux nécessaires pour réaliser des extensions vidéo et audio sont disponibles sur ce connecteur.

PERSND Ce signal est relié directement à la sortie "son" de l'EXL 100. Il peut être utilisé comme une entrée (carte musicale par exemple), ou comme une sortie (vers un amplificateur).

VIDEO C'est le signal vidéo provenant d'un téléviseur.

SLL Signal de synchronisation lignes. C'est une entrée si le contrôleur d'écran utilisé en mode "Synchro Externe" (Pour réaliser des incrustations, par exemple), et une sortie dans le cas contraire.

SCT Signal de synchronisa-tion composite (ou trames). Mêmes remarques que pour SLL. Lorsque SCT est une entrée, il peut recevoir soit un signal synchro composée, soit un signal de synchro trames.

DOTCLK C'est l'horloge "points" du contrôleur d'écran. Elle permet d'utiliser un crayon optique, ou un système de digitalisation d'images.

CL C'est une entrée qui va commander le niveau du signal de commutation lente de la prise péritélévision. CL doit être à zéro ou non connecté pour que le signal de commutation soit à +12 Volts.

CLSTAT C'est une entrée directe du microprocesseur TMS 7020 (bit 5 du port A) qui est utilisée par le BASIC et d'autres logiciels pour vérifier que le signal CL est bien passé à 1 lorsque l'on désire passer en mode "incrustation".

C'est un moyen de vérifier que la carte d'extension en place comporte bien les circuits d'extraction de la synchronisation.

7. Le bus DMA

Il existe sur le contrôleur vidéo (VDP) de l'EXL 100 un bus DMA, fonctionnant seulement en entrée. Ce bus a été conçu au départ pour la réception de données ANTIOPE, mais il est aussi utilisable pour d'autres applications: Digitalisation d'image, lecture de CD ROMs (compact disc).

Rappelons que la technique DMA (Direct Memory Access) permet de lire ou d'écrire dans la mémoire sans passer par le microprocesseur. Elle est indispensable lorsque le débit des données est très élevé.

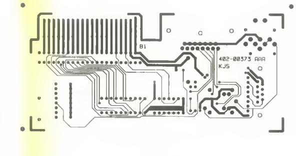
Le bus DMA se compose de deux lignes HMP et HIZ, et de 8 lignes de données (D0 à D7). Lorsque le périphérique veut mettre un octet dans la mémoire, il envoie un niveau 0 sur l'entrée HMP. Lorsque le VDP est prêt à recevoir les données, les signaux HIZ et CAS sont à 0, et le périphérique peut envoyer des données sur les lignes D0 à D7. Le périphérique doit alors repositionner le signal HMP au niveau 1.

Tous les détails concernant l'utilisation de ce bus sont contenus dans la notice du VDP de Texas Instruments (TMS 3556 Video Display Processor : User's manual).

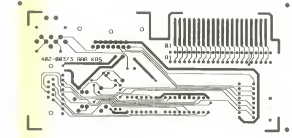
Broche	Signal	Entrée / Sortie
A1	PEREN1	S
A2	PEREN2	S
A3	RSTB	S
A4	RXD	E
A5	TXD	S
A6	N.P.U	
A7	DTR	E
A8	DSR	S
A9	SCLK	E/S

Broche	Signal	Entrée / Sortie
A9	SCLK	E/S
A10	SLL	E/S
A11	-12 V	
A12	DB7	E/S
A13	DB6	E/S
A14	DB5	E/S
A15	DB4	E/S
A16	DB3	E/S
A17	DB2	E/S
A18	DB1	E/S
A19	DB0	E/S
A20	ENABLE	S
A21	ALATCH	S
A22		
A23	CL	S
A24	PERSND	E/S
A25	+12 V	
B1	0V	
B2	+ 5 V	
B3	PERINT	E
B4	PERPRES	E
B5	CLOCKOUT	S
B6	N.P.U	
B7	CLSTAT	E
B8	R / W	S
B9	CAS	S
B10	VIDEO	S
B11		
B12	SCT	E/S
B13	HIZ	S
B14	HMP	E
B15	D7	E
B16	D6	E
B17	D5	E
B18	D4	E
B19	D3	E
B20	D2	E
B21	D1	E
B22	D0	E
B23		
B24		
B25	-5 V	

N.P.U = ne pas utiliser



EXELVISION
TELECOMMANDE MAGNETO +
INTERFACE SOURIS



EXELVISION
TELECOMMANDE MAGNETO +
INTERFACE SOURIS

Figure 3: schéma de la carte 8 entrées-8 sorties

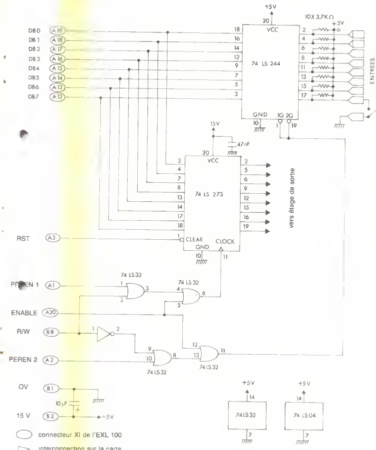


Figure 4: **utilisation d'un relais pour contrôler un témoin lumineux**
Exemple d'étage de sortie

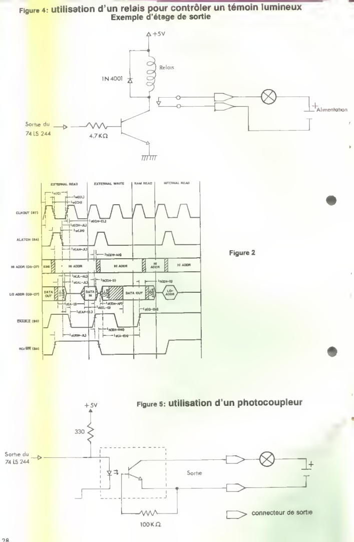


Figure 2

Figure 5: **utilisation d'un photocoupleur**

Les astuces des programmeurs

Le premier listage décrit un sous-programme permettant le tri d'un tableau. Le deuxième listage vous rendra bien des services. Vous avez sans doute remarqué que lorsque vous utilisez le générateur semi-graphique, il vous de faire un INPUT ou un ACCEPT, même restreint en longueur, sans perturber la totalité des 80 caractères qui suivent le début de la saisie. Ce petit sous-programme peut se substituer à un ACCEPT ou un INPUT en permettant d'effectuer une saisie formatée sans que l'environnement semi-graphique du champ de saisie soit perturbé. En outre, la saisie est accentuée et conserve les fonctions CTL S, CTL E, TAB, HOME, DEL, CTL A, etc... Il sera possible de paramétrer les couleurs caractère/ fond, de filtrer les caratères saisis, et de supprimer les fonctions superflues.

Le troisième listage permet de saisir un texte en caratères curvilignes et de l'éditer sur imprimante.

```
100 ! SS PGM DE TRI ALPHABETIQUE D UN TABLEAU NOM$() DE N ELEMENTS M.B 25/06/86
110 !
120 DIM NOM$(30)
130 CLS  WEE  CALL COLOR("DW")
140 CLS
150 LOCATE (5,5)
160 INPUT  NOMBRE DE NOM A SAISIR  ? ":N
170 FOR I=1 TO N
180 LOCATE (10,1) PRINT  NOM NRO ";I;
190 INPUT NOM$(I)
200 NEXT
210 CALL TRI(NOM$(),N)
220 FOR I 1 TO N PRINT NOM$(I) NEXT
230 END
240 !
250 SUB TRI(NOM$() N)
260  DETERMINATION LONGUEUR MAX A TRIER
270 MAX 0
280 FOR I 1 TO N
290 L=LEN(NOM$(I)) IF L>MAX THEN MAX=L
300 NEXT
310 TRI=0
320 FOR I=1 TO N 1
330 P=0 A$ NOM$(I) B$=NOM$(I+1)
340 P=P+1 IF P>L THEN 490
350 CH1 ASC(SEG$(A$ P 1)) CH2=ASC(SEG$(B$ P 1))
360 IF P>1 THEN 410
370 ! TRAITEMENT SI FICHE VIDE
380 IF CH1 32 AND CH2 32 THEN 490
390 IF CH1 32 THEN 480
400 ! TRAITEMENT RECURSIF
410 IF CH1 32 AND CH2 32 THEN 340
420 IF CH2<CH1 THEN 480
430 IF CH2 32 THEN 490
440 IF CH2>CH1 THEN 490
450 ! RETOUR POUR APPROFONDISSEMENT
460 GOTO 340 ! RETOUR APPROFONDISSEMENTDEL 290
470 ! PERMUTATION ET MAJ ORDRE
480 NOM$(I)=B$ NOM$(I+1)=A$ TRI 1
490 NEXT I
500 IF TRI=0 THEN 510 ELSE 310
510 SUBEND
```

```
10 ! SAISIE FORMATEE ACCENTUEE AVEC ENVIRONNEMENT SEMI GRAPHIQUE M B 15/05/86
110 DIM BUF(80)
120 CLS FOR I 1 TO 3
130 CALL ACC(10,1,40,CH$ BUF())
140 LOCATE (15+2xI 1) PRINT CH$
150 NEXT END
160 SUB ACC(Y,X LONG,CH$,BUF())
170 IF LONG+X<-40 THEN 190
180 LONG 41 X
190 FOR I=0 TO LONG 1 BUF(I) 32 NEXT
200 LOCATE (Y X) CALL COLOR( DWF") PRINT RPT$("  LONG)
210 I 0
220 IF I 0 THEN I=0
230 IF I>LONG 1 THEN I LONG 1
240 LOCATE (Y X+I) CALL COLOR( DFW") PRINT CHR$(BUF(I))
250 CALL KEY(A B) IF A 255 THEN 250
260 IF B 1 THEN 280
270 CT=CT+1 IF CT>5 THEN 290 ELSE 250
280 CT 0 CALL SPEECH( L,007C0BAAEA0602FC )
290 CALL COLOR( DWF   LOCATE (Y X+I)
300 IF A 13 THEN 430
310 IF A 132 THEN 400
320 IF A 131 THEN 410
330 IF A 129 THEN 420
340 IF A 13 THEN 440
350 IF A=5 THEN 190
360 IF A=8 THEN 470
370 IF A=1 THEN 510
380 IF A=3 THEN 550
390 BUF(I) A PRINT CHR$(A) I I+1 GOTO 220
400 PRINT CHR$(BUF(I)) GOTO 210
410 PRINT CHR$(BUF(I)) I I 1 GOTO 220
420 PRINT CHR$(BUF(I)) I I+1 GOTO 220
430 PRINT CHR$(BUF(I)) GOTO 560
440 IF I=LONG 1 THEN 240
450 FOR S I TO LONG 1 BUF(S)=32 NEXT
460 LOCATE (Y X+I) PRINT RPT$("  LONG I) GOTO 240
470 LOCATE (Y X+LONG 1) PRINT
480 FOR J=I TO LONG 2 BUF(J)=BUF(J+1) NEXT BUF(LONG 1) 32
490 FOR J=I TO LONG 2 LOCATE (Y,X+J) PRINT CHR$(BUF(J)) NEXT
500 GOTO 240
510 LOCATE (Y X+I) PRINT
520 FOR J=LONG 2 TO I STEP  1 BUF(J+1)=BUF(J) NEXT BUF(I)=32
530 FOR J=I TO LONG 1 LOCATE (Y X+J) PRINT CHR$(BUF(J)) NEXT
540 GOTO 240
550 CALL COLOR( DWF ) PRINT CHR$(BUF(I)) I (INT(I/10)+1)x10 GOTO 230
560 CH$   FOR S=0 TO LONG 1 CH$=CH$&CHR$(BUF(S)) NEXT
570 SUBEND
```

```
100 ! SAISIE FORMATEE ACCENTUEE AVEC ENVIRONNEMENT SEMI GRAPHIQUE M B 15/05 86
110
120 DIM BUF(40)
130 CLS FOR I=1 TO 3
140 CALL ACC(10 1,40,CH$,BUF())
150 LOCATE (15+2xI 1) PRINT CH$
160 NEXT END
170 SUB ACC(Y,X,LONG CH$,BUF())
180
190 ! LIMITATION LONG
200 IF LONG+X<-40 THEN 240
210 LONG=41 X
220 !
230 ! NETTOYAGE BUFFER DE SAISIE
240 FOR I=0 TO LONG 1 BUF(I) 32 NEXT
250
260 ! AFFICHAGE CHAMPS DE SAISIE
270 LOCATE (Y X) CALL COLOR( DWF ) PRINT RPT$("  " LONG)
280
290 ! INITIALISATION 1ER CARACT  A SAISIR
300 I=0
310 !
320 ! SUJETS DE CORRECTION POSITION CARACTERE
330 IF I<0 THEN I=0
340 IF I>LONG 1 THEN I=LONG 1
350 !
360 ! AFFICHAGE CURSEUR SUR CONTENU BUFFEUR
370 LOCATE (Y X+I):CALL COLOR("DFW") PRINT CHR$(BUF(I))
380
390 ! SAISIE D UNE TOUCHE
400 CALL KEY(A B)::IF A=255 THEN 400
410
420 ! TEST SI TOUCHE   TETE DE SERIE
430 IF B=1 THEN 490
440 !
450 ! REJET TEMPORAIRE AVANT AUTO REPEAT
460 CT=CT+1 IF CT>5 THEN 520 ELSE 400
470 !
480 ! REINITIALISATION COMPTEUR POUR REJET EVENTUEL + TIC
490 CT=0 CALL SPEECH("L,007C0BEAAEA0602FC")
500
510 ! REPOSITIONNEMENT PREVISIONNEL DU CARACTERE SAISI
520 CALL COLOR  DWF") LOCATE (Y X+I)
530
540 IF A=13 THEN 700 !     ENVOI
550 IF A 132 THEN 670 !    HOME
560 IF A 131 THEN 680      (
570 IF A=129 THEN 630::!    )
580 IF A=19 THEN 730 !   CTL S (SUPP)
590 IF A=5 THEN 240::!   CTL E (CLR)
600 IF A=8 THEN 790 !    DEL
610 IF A 1 THEN 850 !   CTL A (INSERT) !
620 IF A=9 THEN 910 !     TAB
630 !
640 ! MAJ BUFFER + AFFICHAGE CARACT  + INCREM  POSITION CURSEUR
650 BUF(I)=A PRINT CHR$(A) I I+1 GOTO 330
660 !
670 PRINT CHR$(BUF(I)) GOTO 300
680 PRINT CHR$(BUF(I)) I I 1 GOTO 330
690 PRINT CHR$(BUF(I)) I I+1 GOTO 330
700 PRINT CHR$(BUF (I)) GOTO 940
710
720 ! TRAITEMENT SUP
730 IF I>LONG 1 THEN 370
740 FOR S=I TO LONG 1 BUF(S) 32 NEXT
750
760 LOCATE (Y X+I) PRINT RPT$("  " LONG I) GOTO 370
770
780 ! TRAITEMENT DEL
790 LOCATE (Y,X+LONG-1) PRINT
800 FOR J=I TO LONG 2 BUF(J)=BUF( +1) NEXT BUF(LONG 1) 32
810 FOR J=I TO LONG 2 LOCATE (Y X+J) PRINT CHR$(BUF(J)) NEXT
820 GOTO 370
830 !
840 ! TRAITEMENT INSERTION
850 LOCATE (Y,X+I) PRINT "
860 FOR J=LONG 2 TO I STEP  1 BUF(J+1)=BUF(J) NEXT BUF(I)=32
870 FOR J=I TO LONG 1 LOCATE (Y X+J) PRINT CHR$(BUF(J)) NEXT
880 GOTO 370
890 !
900 ! TRAITEMENT TAB
910 CALL COLOR( DWF  ) PRINT CHR$(BUF(I)) I=(INT(I 10)+1)x10 GOTO 340
920
930 ! CONVERSION BUFFER  + CHAINE
940 CH$   FOR S 0 TO LONG 1 CH$=CH$&CHR$(BUF(S)) NEXT
950
960 SUBEND
```

Editeur de texte

Avec ce programme, il vous est possible d'écrire en cursive, en utilisant le clavier de votre ordinateur.

Courrier, convocations, publicités diverses, seront traités d'une manière très originale.

Voilà une adaptation de plus, au programme de M. Paul Picard, paru dans le numéro 3 d'Excelement Vôtre.

Cela devrait mettre en valeur les possibilités de l'Imprimante E X L.

```
10 !   --  ---  -   -- -----  -------- ------------------ ----------
11  ***        EDITEUR DE TEXTE
12 !-  -   -----  ---- -- ---------  -------  --------  --------------
15 ! Ce programme reprend l'Editeur de  Texte de M. Paul FICARD, publie p 7
20 ! dans EXELEMET VOTRE No 3
30   MAIS, cette fois, les caracteres   vont apparaitre en CURSIVE.
40 !  ---   - -----  ------ ------  ------ ------- --------- ------ --
45 ! Adaptation realisee
50 ! par M. HABERT, instituteur
55 ! a ARQUES (62510)
60 !   - -    -  ----- ------   ------ ----- ------  ----------------- -
130 !*** Commandes :
140 !                 - RETURN
150 !                   TAB
160 !                   SKIP [ Fonction S ]
170 !                 - CLS  [ Fonction C ]
180 !                   HOME
195 !          EXPLICATIONS [ Fonction A ]
200 !  Deplacement du curseur dans les quatre directions
210    Impression de l'ecran sur l'imprimante : CURSIVE        [ Fonction [ ]
220 !                             - -   -  ---   ----- --  ----      --
230 CLS  CFF
240 CALL COLOR( OGF )
250 CALL POKE(51000,138,194,150,182,138,194,149,152,1,14,124,41,14)
260 CALL POKE(51013,210,13,56,13,41,0,88,255,41,53,176,207,131,152,1,14)
270 CALL POKE(51029,246,215,0,35,131,37,4,130,46,10)
280 ATTACH RETC,SKIP
290 LOCATE (1,1):IS=0
300 !           -  Saisie ecriture______
310 CALL EXEC(51000)
320 CALL KEY1(K,L)
330 IF L 0 THEN F=0:GOTO 320
340 IF L= 1 THEN F=F+1:IF F<12 THEN 320 ELSE 370
350 IF L=1 THEN IS=9
360 CALL SPEECH( L,0D7CBEAAEAE602FC )
370 CALL EXEC(51000)
380 ON IS GOTO 580,590,600,610,620,630,640,650,390
390 IF K>131 OR K<128 THEN 490
400 ON K 127 GOTO 410,420,430,440
410 IS=1:V= 1:GOTO 450
420 IS=2:H=1:GOTO 450
430 IS=3:V=1:GOTO 450
440 IS=4:H= 1:GOTO 450
450 GOSUB 670
460 V=0
470 H=0
480 GOTO 310:!    Fin gestion curseur _
490 IF K=13 THEN CALL RETC:IS=5:GOTO 310
500 IF K=133 THEN CALL EXPL:IS=9:GOTO 310
510 IF K=197 OR K=223 THEN CALL CLS:GOTO 310
520 IF K=133 THEN LOCATE (1,1):GOTO 310
530 IF K=211 OR K=243 THEN CALL SKIP:IS=6:GOTO 310
540 IF K=201 THEN CALL TMP:IS=9:GOTO 310
550 IF K=212 OR K=244 THEN GOSUB 750:IS=7:GOTO 310
570 PRINT CHR$(K);:IS=8:GOTO 310
580 V=-1:GOTO 450
```

```
590 H=1:GOTO 450
600 V=1:GOTO 450
610 H=-1:GOTO 450
620 CALL RETC.GOTO 110
630 CALL SKIP:GOTO 310
640 GOSUB 750:GOTO 310
650 PRINT CHR$(K);:GOTO 310
660 !--        Deplacement  du curseur
670 CALL PEEK(43813,COL,LIG)
680 IF COL=40 THEN CALL RETC:GOTO 730
690 IF LIG>21 THEN IF V=1 THEN 730
700 IF LIG<=1 THEN IF V=-1 THEN 730
710 IF COL<=1 THEN IF H=-1 THEN 730
720 LOCATE (LIG+V,COL+H)
730 RETURN
740 !- --    ----  TAB - -------------- -
750 CALL PEEK(43813,COL,LIG)
760 COL=1+10*INT((COL+10)/10)
770 IF COL>40 THEN COL=1:LIG=LIG+1:IF LIG>22 THEN 790
780 LOCATE (LIG,COL)
790 RETURN
800 SUB RETC!---  -- Retour charzot___
810 CALL PEEK(43813,COL,LIG)
820 IF LIG<22 THEN 840
830 CALL POKE(43813,1,23):PRINT RPT$(" ",39):LIG=LIG-1
840 LOCATE (LIG+1,1)
850 SUBEND
860 SUB CLS! -   -  CLS  ----------
870 CALL PEEK(43813,COL,LIG)
880 LOCATE (LIG,1)
890 FOR I=1 TO 23 LIG:PRINT RPT$(" ",40);.NEXT
900 LOCATE (LIG,COL)
910 SUBEND
920 SUB SKIP     ---- Skip  - -    ---
930 CALL PEEK(43813,COL,LIG)
940 PRINT RPT$(" ",41-COL)
950 LOCATE (LIG,COL)
960 SUBEND
1120 SUB MOD(A$)
1130 FOR I=1 TO 40
1140 B$=SEG$(A$,I,1):IF B$=CHR$(0)THEN B$=" "
1150 C$=C$&B$
1160 NEXT
1170 A$=C$
1180 SUBEND
1190 SUB IMP!-  Impression Cursive --
1200 CALL PEEK(43813,COL,LIG)
1210 DIM L$(3,172)
1220 K1$=CHR$(123):K2$=CHR$(124):K3$=CHR$(125):K4$=CHR$(126):K5$=CHR$(127)
1230 P$=CHR$(96):Q$=RPT$(K5$,6):EL$="@@@@@
1240 RI$=EL$&BL$
1250 FOR N=1 TO 172:FOR I=1 TO 3:L$(I,N)=RI$:NEXT I.NEXT N
1260 L$(1,2)="@@@DEE@@@@":L$(2,2)="HN_aaa'&K4$&K5$&"AF'
1270 L$(1,15)="@@@E@@E@@@":L$(2,15)=L$(2,2)
1280 L$(1,16)="@@@DEE@@@@":L$(2,16)="HN_iirBDD"
1290 L$(1,17)="@@@EFD@@@@":L$(2,17)=L$(2,16)
1300 L$(1,18)="@@@BCUD@@@":L$(2,18)=L$(2,16)
1310 L$(2,20)="HN_aaaBEDD":L$(3,20)="@@@LTH@@@@"
1320 L$(1,21)="@@@E@PE@@@":L$(2,21)=L$(2,16)
1330 L$(1,26)="@@E@@F@@@@':L$(2,26)="HHF"&K4$&K5$&'AAEDD"
1340 L$(1,27)='@@EDFE@@@@":L$(2,27)="HN_aaq"HDD"
1350 L$(1,33)="@@@@@U@@@@":L$(2,33)="@@@@@y@@@@"
1360 L$(1,39)="@@@@FX@@@@"
1370 L$(2,44)='@@@@@C@@@@':L$(3,44)="@@@@p@@@@@'
1380 L$(2,45)="@@LLLLLL@@"
1390 L$(2,46)="@@@@CC@@@@'
1400 L$(1,48)='@AGFDELD@:L$(2,48)='@"&K4$&K5$& AAA &K4$&"@":L$(3,48)="@@@@@@@@'
1410 L$(1,49)='@@AEGG@':L$(2,49)="@"&K$&'@@"&K5$&K5$&'@:L$(3,49)='@@@@@@@'
1420 L$(1,50)='@LLLGL@":L$(2,50)="@co"&K3$&'qA@":L$(3,50)=L$(3,49)
1430 L$(1,51)=L$(1,50):L$(2,51)='@FQyoF@:L$(3,51)=L$(3,49)
1440 L$(1,52)="@@CGGG@@:L$(2,52)="@ppF'&K5$&K5$&'F@":L$(3,52)="@@@@@@@@'
1450 L$(1,53)="@CGDUD@':L$(2,53)="@rqS L@":L$(3,53)=L$(3,49)
1460 L$(1,54)="@ACFDDL@:L$(2,54)='@ &K2$&K4$&'SQQN@':L$(3,54)=L$(3,48)
1470 L$(1,55)='@UDDFC@:L$(2,55)="@CO &K2$&'p@@':L$(3,55)=L$(3,49)
1480 L$(1,56)="@CGDUC@:L$(2,56)="@fyY_f@":L$(3,56)=L$(3,49)
1490 L$(1,57)="@LUDFCA@:L$(2,57)="@fQQS"&K4$&K2$&"@":L$(3,57)=L$(3,54)
1500 L$(2,58)='@@@@@T@@@@
1510 L$(2,59)='@@@@@K@@@@":L$(3,59)=L$(3,44)
1520 L$(1,63)="@@LUHHDL@@":L$(2,63)="@@@@@Mp@@@'
1530 L$(1,65)='@@@DF"&K5$&K5$& A@@":L$(2,65)='FAa"&K4$&K$&K4$&K5$&"AIF'
1540 L$(1,66)="NQR'&K$&'go'&K$& s"L@":L$(2,66)="DEAC"&K4$&K2$&"@fq_N"
1550 L$(3,66)=K2$& @"
1560 L$(1,67)='@LRGDYqaRL":L$(2,67)="@@@"&K2$&K4$&"CAEIF
1570 L$(1,68)="@LR &K$& go'&K$&'THG':L$(2,68)="@FIK"&K4$&K2$&"SAB"&K2$
1580 L$(1,69)='@@PXJN_chF':L$(2,69)='@@@N_yqaJD"
1590 L$(1,70)="NQh'&K$& go &K$&"QRa":L$(2,70)='DBQc"&K4$&K2$&"dbdH"
1600 L$(1,71)='LRGOYqahL@":L$(2,71)="@@"&K2$&K4$&"CAADDD"
1610 L$(3,71)='@@@FIQa &K4$&K2$&"@"
1620 L$(1,72)="xF &K$&K5$&K5$&'@_"&K5$&"bdX"
1630 L$(2,72)="FIQ"&K5$&K4$&"@"&K4$&K5$&"AIF":L$(3,72)=RI$&"@"
1640 L$(1,73)='@@@xFF"&K$&K5$&K5$&'@':L$(2,73)='@@@DEAA"&K5$&K4$&'P"
```

```
1650 L$(1,74)="@@xPP"&K$&K5$&K5$&"@@":L$(2,74)="@@@@@A"&K5$&K5$&"HP"
1660 L$(3,74)="@@NQ@A"&K4$&K2$&"@@"
1670 L$(1,75)="xP"&K$&K5$&K5$&"@_"&K5$&K$&"dX"
1680 L$(2,75)="FAA"&K5$&K4$&K$&K5$&"_AIF":L$(3,75)=RI$&"@"
1690 L$(1,76)="@LRA_"&K5$&"aRL@":L$(2,76)="@BEE"&K5$&K4$&"AEIF"
1700 L$(1,77)="@@OP"&K5$&K5$&"@OP"&K5$&K5$&"@@@"
1710 L$(2,77)="FA"&K4$&"@"&K4$&K5$&"A"&K4$&"@"&K4$&K5$&"AIF":L$(3,77)=RI$&"@@@@"
1720 L$(1,78)="@@OP"&K5$&K5$&"@OFb\"
1730 L$(2,78)="FA"&K4$&"@"&K4$&K5$&"A"&K4$&"@@@":L$(3,78)=RI$&"@"
1740 L$(1,79)="@@GOXFdhHG":L$(2,79)="@@"&K2$&K4$&"CAAA@"&K2$
1750 L$(1,80)="NQR"&K$&"go"&K$&"dRL":L$(2,80)="DBAA"&K5$&K4$&"OOHH"
1760 L$(1,81)="GOXpdbH"&K5$&K5$:L$(2,81)=K2$&K4$&"CAAA@"&K5$&K5$
1770 L$(3,81)="@@@@@@@"&K5$&K5$
1780 L$(1,82)="NQR"&K$&"go"&K$&"s~L@@@"
1790 L$(2,82)="DBAC"&K4$&K2$&"@"&K$&K4$&"_AIF":L$(3,82)=RI$&"@@@"
1800 L$(1,83)=L$(1,75):L$(2,83)="@FIA"&K5$&K4$&"@@@@"
1810 L$(1,84)="BEIK_"&K3$&"xJP@":L$(2,84)="@@@x"&K4$&"GAIRL"
1820 L$(1,85)="Xd"&K$&K5$&"_@@@"&K5$&K5$&"@@"
1830 L$(2,85)="@@@"&K4$&K5$&"AAE"&K4$&K5$&"AF":L$(3,85)=RI$&"@@"
1840 L$(1,86)="Xd"&K$&K5$&"_@XdTOO":L$(2,86)="@@@"&K2$&K4$&"CAAC"&K2$&"D"
1850 L$(3,86)=RI$&"@"
1860 L$(1,87)="Xd"&K$&K5$&"_@"&K5$&K5$&"@Fh_D"
1870 L$(2,87)="@@@"&K2$&K5$&"A"&K5$&K5$&"AA@"&K2$&"D":L$(3,87)=RI$&"@@@"
1880 L$(1,88)="\bx_GAG_yb\":L$(2,88)="NGg"&K4$&"x@x"&K4$&"GQN"
1890 L$(3,88)=RI$&"@"
1900 L$(1,89)="Xd"&K$&K5$&"_@@@"&K5$&K5$&"@"
1910 L$(2,89)="@@@"&K4$&K5$&"AA@"&K5$&K5$&"D"
1920 L$(3,89)="@@@@FIQa"&K4$&K2$&"@"
1930 L$(1,90)="xPFco"&K3$&"q@":L$(2,90)="co"&K3$&"rCCDH"
1940 L$(3,90)="FIQcD"&K2$&"p@"
1950 L$(2,95)="@@CCCCCC@@"
1960 L$(2,97)="HN_aaa"&K4$&K5$&"AF"
1970 L$(1,98)="@@O_F"&K$&"a~@@":L$(2,98)="HH"&K4$&K5$&"QaINDD"
1980 L$(2,99)="HN_aaa@DDD"
1990 L$(1,100)="@@@@@@@DD@@":L$(2,100)=L$(2,97)
2000 L$(2,101)="HH_iiir@DD"
2010 L$(1,102)=L$(1,98):L$(2,102)="HH"&K5$&K5$&"Ra@DDD"
2020 L$(3,102)="@@"&K2$&K4$&"BAa~@@"
2030 L$(1,103)=RI$:L$(2,103)="HN_aaa"&K5$&K5$&"@D"
2040 L$(3,103)="@@FIQa"&K4$&K2$&"@@"
2050 L$(1,104)=L$(1,98):L$(2,104)="HH"&K5$&K5$&"Pp~DAF"
2060 L$(1,105)="@@@LL@@@@@":L$(2,105)="HHP"&K4$&K5$&"AA@DD"
2070 L$(1,106)="@@@@@LL@@@":L$(2,106)="HHHPP"&K5$&K5$&"@DD"
2080 L$(3,106)="@FIQa"&K4$&K2$&"@@@"
2090 L$(1,107)=L$(1,98):L$(2,107)="HH"&K5$&K5$&"Tt~KAF"
2100 L$(1,108)=L$(1,98):L$(2,108)="HH"&K4$&K5$&"QaA@DD"
2110 L$(1,109),L$(3,109)=RI$&"@@"
2120 L$(2,109)="X"&K$&K5$&"_"&K$&K5$&"_"&K$&K4$&"_AF"
2130 !
2140 L$(2,110)="X"&K$&K5$&"_"&K$&K$&K4$&"_RF"
2150 L$(2,111)="HN_aaq~HDD"
2160 L$(2,112)="HP"&K5$&K5$&"P"&K$&K4$&"_AF"
2170 L$(3,112)="@@"&K5$&K5$&"@@@@@@"
2180 L$(2,113)="HN_aaa"&K5$&K5$&"@D"
2190 L$(3,113)="@@@@@@"&K5$&K5$&"@@"
2200 L$(1,114)="@@A@@@@@@@":L$(2,114)="HP"&K$&K$&K4$&"_A@DD"
2210 L$(1,115)="@@@@@A@@@@":L$(2,115)="HDDNRay_FD"
2220 L$(1,116)="@@@DDE@@@@":L$(2,116)="HP"&K$&K4$&K5$&"AA@@D"
2230 L$(2,117)="HP"&K4$&K5$&"A@"&K4$&K5$&"AF"
2240 L$(2,118)="HP"&K$&K4$&"_AAr\H"
2250 L$(1,119),L$(3,119)=RI$&"@@":L$(2,119)="HP"&K$&K4$&"_A"&K4$&K5$&"Ar\H"
2260 L$(2,120)="HRa@"&K4$&"_saRD"
2270 L$(2,121)="HP"&K$&K4$&"_AA"&K5$&K5$&"D"
2280 L$(3,121)="@@@FIQa"&K4$&K2$&"@"
2290 L$(1,122)="@@@A@@D@@@":L$(2,122)="HHYcfoyrDD"
2300 L$(3,122)="@@FIQa"&K4$&K2$&"@@"
2310 OPEN #1,"20",OUTPUT
2320 CALL POKE(3,13):PRINT #1
2330 FOR I=1 TO 22
2340 LOCATE (I,1)
2350 ACCEPT SIZE(-40),A$
2360 CALL MOD(A$)
2370 PRINT #1,CHR$(28);
2380 FOR T=1 TO 3:PRINT #1,RPT$("@",80);:FOR J=1 TO LEN(A$):LT$=SEG$(A$,J,1)
2390 G=ASC(LT$):PRINT #1,L$(T,G);:NEXT J:PRINT #1,"6":NEXT T
2400 NEXT I
2410 PRINT #1,"O";CHR$(29)
2420 CALL POKE(3,4)
2430 CLOSE #1:LOCATE (LIG,COL)
2440 SUBEND
2450 SUB EXPL!________ Explications ___
2460 CALL SKIP
2470 PRINT "taper :        Ecran      Imprimante"
2480 !
2490 PRINT "Ctl C :            ";CHR$(27);"           ";CHR$(27)
2500 PRINT "Ctl R :            ";CHR$(18);"           ";CHR$(18)
2510 PRINT "Ctl U :            ";CHR$(21);"           ";CHR$(21)
2520 PRINT "Ctl Z :            ";CHR$(26);"           ";CHR$(26)
2530 !
2540 PRINT "Ctl O :            ";CHR$(15);"           ";CHR$(15)
2550 SUBEND
```

EXELEMENT VOTRE N° 10

sommaire

Juin-Juillet 1986
REVUE BIMENSUELLE
Prix: 20 F

REDACTION: Patrice Chaillan

ONT COLLABORE: à la conception de ce numero, toute l'equipe d'EXELVISION de SOPHIA

ILLUSTRATION: Catherine Chaillan

PHOTOS: Patrice Chaillan

DIRECTEUR de la PUBLICATION: Exelvision

RELATIONS LECTEURS:
Exelvision - Courrier Exelement Vôtre - Place Joseph Bermond
Immeuble Ophira - 06560 Valbonne

ABONNEMENTS: 6 numeros
120 F, frait de port 3 F/numéro
Règlement e l ordre d'Exelvision

PUBLICITE: Ecrire au journal et envoyer textes ou typons au format, au plus tard 3 semaines avant parution

DIFFUSION: points de vente Exelvision

COUVERTURE: Masse

Imprimé par
l'Atelier d'Impression